

Review

Open-Source Electronics Platforms as Enabling Technologies for Smart Cities: Recent Developments and Perspectives

Daniel G. Costa ^{1,*}  and Cristian Duran-Faundez ^{2,†} 

¹ Department of Technology, State University of Feira de Santana, Feira de Santana 44036900, Brazil

² Department of Electrical and Electronic Engineering, University of the Bío-Bío, Concepción 4051381, Chile; crduran@ubiobio.cl

* Correspondence: danielgcosta@uefs.br; Tel.: +55-75-992389000

† These authors contributed equally to this work.

Received: 18 November 2018; Accepted: 5 December 2018; Published: 7 December 2018



Abstract: With the increasing availability of affordable open-source embedded hardware platforms, the development of low-cost programmable devices for uncountable tasks has accelerated in recent years. In this sense, the large development community that is being created around popular platforms is also contributing to the construction of Internet of Things applications, which can ultimately support the maturation of the smart-cities era. Popular platforms such as Raspberry Pi, BeagleBoard and Arduino come as single-board open-source platforms that have enough computational power for different types of smart-city applications, while keeping affordable prices and encompassing many programming libraries and useful hardware extensions. As a result, smart-city solutions based on such platforms are becoming common and the surveying of recent research in this area can support a better understanding of this scenario, as presented in this article. Moreover, discussions about the continuous developments in these platforms can also indicate promising perspectives when using these boards as key elements to build smart cities.

Keywords: smart cities; Internet of Things; Raspberry Pi; BeagleBoard; Arduino

1. Introduction

The development of different areas of electronics, computing, data acquisition, and communication has created a fertile environment for the rise of the Internet of Things (IoT) [1]. The IoT is expected to deeply change our lives, allowing the creation of ubiquitous, distributed, and reactive systems that can surprisingly alter the way we interact with the world [2,3]. Smart cities are an important application of the IoT landscape, bringing complex challenges in different aspects of city structure and of their inhabitants' behaviors [4,5], but also opening many commercial, industrial, educational, and cultural opportunities [6–9].

The development of efficient communication technologies and cloud data services with decreasing costs, as well as the affordability of powerful open-source hardware platforms, have boosted smart-city initiatives [10,11]. With such platforms, the development time and related costs are considerable reduced, easing the creation of dedicated smart-city applications that can solve punctual problems. On the other hand, in a bigger scope where smart-city systems are designed and deployed by the governments, embedded hardware platforms are also a feasible solution not only during development phases, but also when providing public data: as most platforms are based on open-source technologies, the adoption of public standards is encouraged. Moreover, academic and scientific smart-city applications are inherently propitious to the adoption of open-source electronics platforms

to construct IoT devices. Such smart-city scenarios may also rise concerns about interoperability and concurrency [12,13], but the characteristics of open-source electronics boards are favorable when promoting compatibility.

Among the existing open-source electronics platforms, some options are becoming more popular due to many factors. In common, the most popular boards share similar characteristics such as being versatile, powerful (although some models are limited) and cost-effective, being useful to create virtually any kind of device [14,15]. In this context, as most of them are ultimately a computer, respecting the different computational capabilities of the boards, they can be used to create sensors, actuators, and any processing device, presenting themselves as wild-cards for different types of sensor networks, IoT and smart-city applications [16,17].

In recent years, open-source electronics platforms have been used to support IoT applications and thus they can also be employed to support the creation of smart cities. From a different perspective, those platforms are also being taken as pillars for applications that are already being developed around the smart-city principles. Therefore, understanding this scenario can also indicate how smart cities are evolving and what we can expect from the near future.

Figure 1 presents a generic example of a smart city where different systems may be implemented using open-source electronics platforms, solving relevant problems of urban environments.

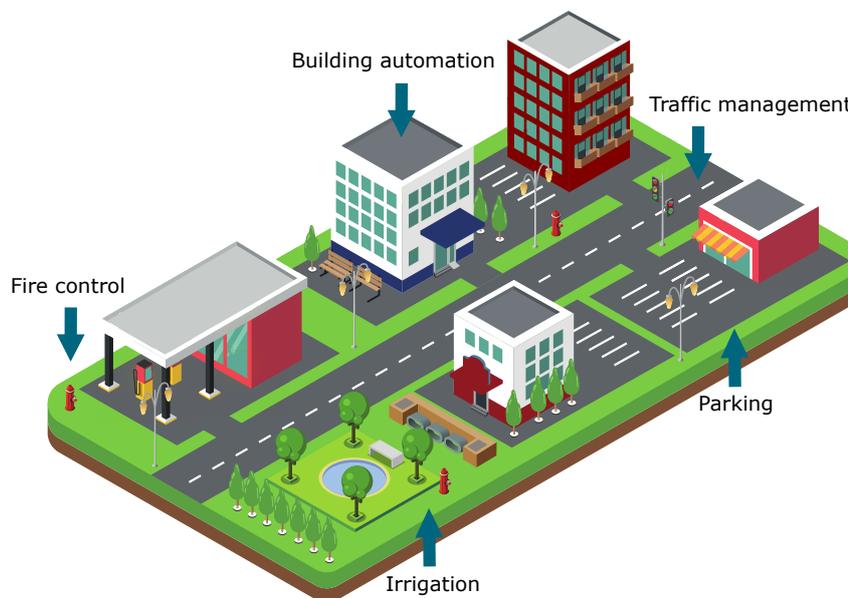


Figure 1. A typical smart city with multiple concurrent systems.

This article surveys recent scientific developments that exploit open-source electronics boards to create IoT and smart-city applications. Moreover, perspectives of this scenario are presented and discussed. As this is a very dynamic area, this article is valuable to support new researches and developments in this field.

The remainder of this paper is organized as follows. Section 2 presents the current availability of embedded open-source platforms that can be used to create IoT devices. Section 3 surveys recent works employing popular open-source boards for IoT and smart-city applications. Perspectives for future developments are presented in Section 4, followed by Conclusions in Section 5 and References.

2. Open-Source Platforms for IoT Development

IoT systems may be designed in different ways, but it is reasonable to expect that sensors, actuators and controller units will be present in most applications. For that, embedded hardware platforms may be used to implement these elements and there are many available options on the market. Among those options, open-source platforms are gaining prominence due to many factors and there are

currently some good boards to be chosen. However, the existence of many research and industrial projects that are developed exploiting a few different boards is a good indication of which are the most popular platforms.

There are some characteristics that must be properly considered when choosing the most appropriate open-source platform for a particular IoT application. Those characteristics are described as follows:

- Low cost: To allow massive deployment, especially for physically scattered applications, the chosen platform must be cheap enough to not compromise the design and maintenance of smart-city systems;
- Computational power: Depending on the desired functionality, the board should have enough computational power to execute complex tasks, such as visual data processing and controlling of multiple distributed nodes. Processing and memory resources directly influence the computational power of the board;
- Programming flexibility: The availability of programming languages and libraries is of paramount importance and this characteristic is not necessarily a function of the available hardware capabilities, but also the relevance of the developers community around the platform;
- I/O interfaces. Smart-city applications may use electronic devices to provide relevant data or to take some action, such as cameras, LED, sirens, LCD panels, sensors, among others. The type and number of I/O interfaces in the adopted boards are important parameters when choosing the platform;
- Low energy consumption: As IoT nodes may be powered by batteries, low energy consumption is a relevant characteristic for embedded hardware platforms. Besides the use of batteries, energy efficiency is highly desired in modern cities, encouraging the use of more efficient boards;
- Reduced size: Depending on the expected functions of the board and the deployment place (e.g., lamp posts, traffic lights, transit plates), the size of the adopted board is relevant and should be considered;
- Robustness: Open-source boards should operate for long periods without encountering a hardware failure condition or overheating. If they are deployed at outdoor areas, such remarks are even more important;
- Communication: Different communication technologies may be embedded into the boards, easing the connections of the nodes. Wi-Fi, Bluetooth, Ethernet, LoRa and 4G/5G cellular standards are common communication technologies that may be present;
- Storage: The existence of embedded permanent storage may be helpful in many scenarios. For some boards, permanent storage may only be implemented through portable memory cards;
- Additional resources: LEDs, buttons, RFID (Radio-Frequency IDentification), required power supply (voltage/current), among other characteristics, may influence the choosing of a platform.

There are many open-source electronics platforms but only some of them are being more frequently used to create IoT applications. Among those boards, we selected some of the most popular platforms to be surveyed, considering affordable prices and large use in IoT applications as decision parameters for such selection. The chosen platforms are described in next subsections.

2.1. Raspberry Pi

Raspberry Pi is an affordable tiny and powerful multi-purpose computer that can be used for a variety of applications. When coming to the IoT and the smart-city environment, Raspberry also performs well as can be seen in the increasing number of projects that are exploiting its potentials. Although other hardware platforms have been created for the same purpose, Raspberry Pi is highly used by the industry, technology companies, governments, universities, and hobbyists, supporting the development of different kinds of applications [18,19].

Taking advantage of new electronic technologies and the success of ARM-based processors, the initial Raspberry Pi board was created with more modest purposes. Using cheap components that

could be combined in a single board, the initial Raspberry Pi board was originally designed to be an affordable computer to assist students in school. However, it rapidly gained interest from other areas due to its flexibility, computational power, and reduced cost, helping it to become one of the best-selling computers in history.

Raspberry Pi is constructed around an ARM (Advanced RISC Machine) processor, which is the basis for the mobile smartphones industry. Therefore, Raspberry not only takes advantage of the characteristics of the RISC (Reduced Instruction Set Computer) processing platform, but also it was designed when manufacturers of RISC processors were consolidated. Since its initial debut, different models and versions of the original Raspberry board were created, addressing different hardware demands.

The increasing success of Raspberry Pi boards is due to many factors, putting this as the leading open-source hardware development platform. In addition, this hegemony has been reinforced by a series of performance evaluations, which has been conducted in recent years, for different scopes. The work in [20] evaluated a Raspberry Pi 2 B board in terms of graphical processing. The CPU and GPU loading was analyzed according to different visual data processing configurations, and energy consumption was assessed for different testing scenarios. From a different perspective, the work in [21] evaluated a Raspberry board (Raspberry Pi B) as a central element for a networking testbed based on the KODO library [22]. Even using an early model of the Raspberry Pi platform, the results in [21] were encouraging. Energy consumption was also evaluated in [23] but considering particular demands when implementing the RESTful Web Services framework. Similarly, the execution of web services in the Raspberry Pi platform was also addressed in [24], which evaluated load balancing in clusters of Raspberry Pi boards. In fact, all those works evaluated the performance of Raspberry Pi boards in different scenarios, indicating the interest of using this platform for a large set of tasks.

The most popular Raspberry Pi boards are the Raspberry Pi 2 B, the Raspberry Pi 3 B, and the Raspberry Zero W. Raspberry Pi was conceived and it is mostly manufactured in UK.

2.2. BeagleBoard

BeagleBoard is an open-source hardware platform that has the same principles of Raspberry Pi and thus they can often be used interchangeably. However, there are some important characteristics that put BeagleBoard as a viable option with its own particularities [25]. As with the Raspberry Pi family, BeagleBoard is a single-board tiny computer with affordable prices and low energy consumption. The initial use of this board and the related projects put both platforms in an equivalent development field.

When designing IoT applications, BeagleBoard has some particularities that can make it the board to be chosen. In a different way to Raspberry Pi, BeagleBoard board models have digital and analog I/O (similarly to the Arduino family). Other helpful characteristic is the presence of an onboard permanent storage unit in the BeagleBone series, which can be used to store code and sometimes to avoid the use of a microSD card, although there is a slot for those cards.

The work in [25] conducted a deep review of the BeagleBoard platform, discussing and evaluating some performance metrics of different board models.

The most popular BeagleBoard boards are the BeagleBone Black and the BeagleBone Blue. BeagleBoard was conceived and it is mostly manufactured in the USA.

2.3. Arduino

Differently from Raspberry Pi and BeagleBoard, the Arduino platform is an embedded prototyping board that is designed for electronics projects but that does not necessarily operate as a computer. Its use is focused on automation and electronic projects that demand repeated execution of some tasks and thus its software and hardware resources are more limited. However, the simplicity of Arduino finds its place in many automation and control projects and it is common to see Arduino boards composing smart-city systems.

While Raspberry Pi and BeagleBoard are developed around an ARM processor, most Arduino models are developed with a simpler micro-controller manufactured by Atmel, with most of them being of the ATmega family (8-bit, against the 32-bit processors of Raspberry Pi and BeagleBone). In addition, this characteristic has two practical implications: first, an Arduino board is usually cheaper; and second, it has less computational power than a usual Raspberry Pi or a BeagleBoard. Moreover, Arduino operates under a very lower clock speed and it has low RAM memory, but there is a flash memory to store the programs, which are written using special software IDE developed by the Arduino manufacturer. High-level programming languages such as C and Python can be used to program Arduino boards.

Although the reduced computational power may sound badly, the Arduino boards can do many things in terms of electronic control and it is a best-selling platform for its purposes. However, it is often limited for many IoT applications, especially when there is intensive computational processing. Despite that, there are a lot of extension boards (*shields*) to enhance the functions of Arduino, adding flexibility and useful resources. Among those extension boards, the ESP8266 family of microchips is a SoC (System on a Chip) that adds communication capabilities to Arduino and it became common to see many Arduino-based projects exploiting popular chips such as the ESP-01 and ESP-12. For IoT applications, such chips allow Arduino to easily communicate through Wi-Fi.

The Arduino platform was designed and manufactured in Italy and some of its most popular boards are the Arduino Uno, the Arduino Mega and the Arduino Due.

2.4. Open-Source Platforms Comparison

There are different options of open-source hardware platforms, with specifications and constraints that must be properly considered when implementing IoT and smart-city applications. The proper choosing of the most appropriate board is not straightforward, leaving to a heterogeneous scenario with projects and applications developed with different boards or employing combinations of them. However, the proper knowledge of the available off-the-shelf boards can give some light to this matter.

The previously presented open-source electronics platforms have been implemented in different models. Table 1 summarizes some of the most popular models of the presented open-source hardware development platforms (the model revisions and upgrades are not displayed). The “networking” column indicates onboard enabled resources, but additional hardware or dongles can be used to add networking capabilities to the boards.

Table 1. Popular open-source hardware platforms for IoT development.

Platform	Release	CPU	RAM	Networking
Arduino Nano	2008	16 MHz	0.5 KB	-
Arduino Uno	2010	16 MHz	2 KB	-
Arduino Mega ADK	2011	16 MHz	8 KB	-
Raspberry Pi B	2012	700 MHz	256 MB	Ethernet
Arduino Due	2012	84 MHz	96 KB	-
BeagleBone Black	2013	1 GHz	512 MB (4GB storage)	Ethernet
Arduino Yun	2013	16 MHz	2.5 KB	Wi-Fi
Raspberry Pi B+	2014	700 MHz	512 MB	Ethernet
Raspberry Pi 2 B	2015	900 MHz	1 GB	Ethernet
BeagleBone Green	2015	1 GHz	512 MB (4GB storage)	Ethernet
Raspberry Pi Zero	2015	1 GHz	512 MB	-
BeagleBoard-X15	2016	1.5 GHz	2 GB (4GB storage)	Ethernet
Raspberry Pi 3 B	2016	1.2 GHz	1 GB	Wi-Fi, Bluetooth/BLE, Ethernet
BeagleBone Blue	2017	1 GHz	512 MB (4GB storage)	Wi-Fi, Bluetooth/BLE
Raspberry Pi Zero W	2017	1 GHz	512 MB	Wi-Fi, Bluetooth/BLE
Raspberry Pi 3 B+	2018	1.4 GHz	1 GB	Wi-Fi, Bluetooth/BLE, Ethernet
Raspberry Pi 3 A+	2018	1.4 GHz	512 MB	Wi-Fi, Bluetooth/BLE

2.4.1. Software Issues

Most of the available open-source hardware platforms will operate as a computer and thus they require an operating system, excepting the Arduino board (the Arduino Yun board comes with a simplified version of the Linux operating system). In addition, there are many different operating systems for those boards, which are more frequently developed under the open-source premises. This creates a healthy environment for innovative projects, which also takes advantage of the average low costs of the embedded open-source hardware platforms.

Some boards do not have onboard storage, such as Raspberry Pi, while BeagleBone, for example, is shipped with an onboard permanent memory unit. When the board has no permanent storage, it usually employs a microSD card to record the operating system image and all user files. In general, terms, operating systems may be differentiated by performance metrics such as portability, required memory, CPU usage, image size, startup time, integrated tools, among others, and thus it is an important decision when designing smart-city applications. Some manufacturers officially support a particular operating system, such as the Raspbian OS for the Raspberry Pi platform, but even Raspberry can use any of dozens of available operating systems. Following the development trend of the open-source platforms, most operating systems for those boards are developed under open-source licenses.

Besides hardware and operating system characteristics, software and middleware are also relevant parameters that have been addressed in some works. The work in [26] discussed different aspects that are relevant when designing IoT applications, including the specification of the services expected from IoT devices and how they can impact the performance of IoT systems. Common services expected for IoT embedded platforms were also discussed in [27]. In that work, services such as virtualization, storage and use of the Rest API are discussed, which are helpful to add flexibility to IoT devices. In a similar way, the work in [28] also discussed services for IoT devices, but with focus on IoT protocols such as CoAP (Constrained Application Protocol) and MQTT (Message Queue Telemetry Transport). In fact, all those works discussed services that should be provided by IoT devices whatever is the employed hardware. Therefore, the presented considerations are valuable when implementing large-scale IoT and smart-city systems.

When choosing the most appropriate solution to construct IoT systems and ultimately smart-city solutions, all the presented aspects should be properly considered, since the particularities of each project should guide the choosing of the hardware and software technologies. Nevertheless, although there are many software middlewares and hardware platforms to develop IoT systems, and companies and research institutions are playing an important role to create this scenario, the existence of generic open-source multi-purpose development hardware platforms is of paramount importance. Open-source boards are revolutionizing the designing, deployment, operation, and maintenance of IoT and smart-city initiatives and thus they deserve more detailed analysis. In addition, in this scenario, the Raspberry Pi and the BeagleBoard family boards are standing out as promising choices, as can be seen in the increasing number of IoT projects based on them. The work in [15] performed interesting comparisons between the Raspberry Pi and the BeagleBoard platforms relating software and hardware issues.

2.4.2. Hardware Issues

In general, words, IoT development may require high flexibility when attaching new components, since different levels of interactions with the environment may be required. In fact, there are many electronic components that can be attached to the chosen open-source boards to extend their functions. For example, sensing units can be attached to allow monitoring of different information such as temperature, pressure, humidity, and pollution. Cameras and display units can also be easily attached. Raspberry Pi, BeagleBoard and Arduino have different I/O interfaces and there are many companies that develop different types of electronic components to be attached onto those boards. Particularly, the availability of GPIO (General Purpose Input/Output) pins allows the connection to a lot of

electronic devices and there are many hardware extensions exploiting them, which significantly facilitates the attachment of new hardware components to the boards. Whatever the case, the number, type, and supported protocols of the I/O pins are important parameters when choosing open-source hardware platforms. In fact, the surveyed boards have different particularities concerning digital and analog I/O pins, which should be properly considered in smart-city projects.

Besides generic I/O, additional interfaces are also relevant, such as the CSI (Camera Serial Interface) standard in the Raspberry Pi platform to allow efficient connection of a camera. Depending on the characteristics of the project, such specialized interfaces may indirectly enhance the performance of the system.

Figure 2 presents three examples of open-source boards. From left to right, the boards are an Arduino Mega ADK, a BeagleBone Black and a Raspberry Pi 3 B.

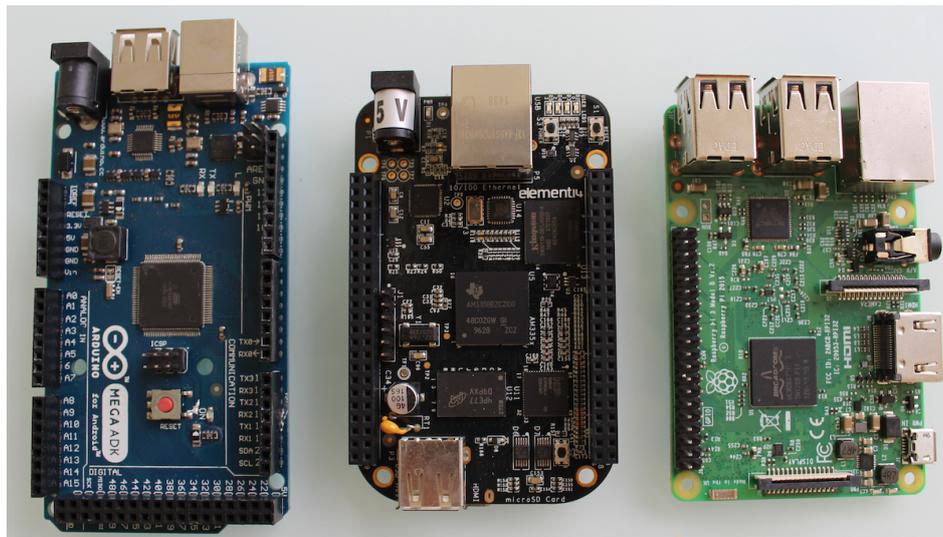


Figure 2. Some popular open-source electronics boards that can be used in IoT and smart-city applications.

3. Smart Cities and Open-Source Electronics Platforms

The availability of open-source prototyping and development platforms has the potential to deeply change the way IoT applications are created and this trend became even more striking with the release of some popular electronics platforms. Although there is a great appeal around these platforms for DIY (Do-It-Yourself) applications, with a lot of projects in the “Smart Home” domain [19,29–31], different types of IoT applications can be designed. Nowadays, the development of smart-city systems based on such platforms is a reasonable and inexpensive choice for many projects. In addition, the last years have given us some clues of what applications can be created and what can still be envisioned for the near future [32].

Open-source electronics platforms will be typically employed to sense information from the environment or to make specialized processing of such information or other data (e.g., received from the Internet). Concerning sensing functions, we can roughly expect that two different types of information can be gathered: scalar and multimedia. Scalar information such as temperature, pressure, humidity, and luminosity is useful for many applications, and open-source boards can be efficiently used to gather and process such sensed data. However, as such data is usually short, the processing, the transmission and the storage of scalar information are not necessarily a critical issue, opening different opportunities for smart-city applications. On the other hand, multimedia sensing will be performed by cameras and microphones to retrieve large amounts of data, which can also have time restrictions when performing real-time processing. Multimedia-based smart cities are thus more challenging, and they rise additional performance concerns. Nevertheless, it will be natural to see some

applications employing scalar and multimedia sensing units at the same board, exploiting continuous developments of more powerful hardware components and more efficient communication standards.

In the context of smart cities, IoT applications are typically designed to solve a particular problem of urban environments. In this sense, smart cities tend to be created by the composition of different concurrent systems [5,33,34]. Considering the state-of-the-art in this area and the use of open-source electronics boards, we selected a group of smart-city common problems to be addressed by such boards, presenting recent works that proposed solutions for them. Although there are many more issues to be addressed by smart-city applications [33], the discussed problems and related approaches are a good indication of the use of open-source boards in this area.

- **Parking system:** Large cities daily face the prominent problem of parking and the increasing number of cars in urban areas is continuously aggravating this issue. IoT applications may then be used to manage free parking spots, indirectly reducing congestion and car accidents when the time trying to find an available spot is reduced;
- **Mobility:** Efficient movement of vehicles, accidents prevention, congestion mitigation and (public) vehicles tracking are some important services than can enhance the mobility in urban areas;
- **Energy efficiency:** Sustainable and efficient consumption of energy in urban areas is one of the challenges of smart-city applications. Most solutions in this area perform dynamical management of the use of energy resources.

The state-of-the-art of IoT and smart-city applications is surveyed in next subsections, but only for works that specify at least one of the considered open-source electronics boards as central elements of the solutions. The selection of the works was made based on this characteristic and the addressing of some of the presented smart-city challenges.

3.1. Parking System

Popular open-source electronics platforms are being used to manage parking systems, following different approaches. The work in [35] employed the Raspberry Pi platform to create an intelligent parking system for smart cities. The authors proposed the use of a Raspberry Pi board to manage the vacant parking spots and to inform drivers about them. Drivers can verify the availability of free parking spots through their smartphones and perform payment when required. A distance (ultrasonic) sensor is used in each parking spot, which is then connected to a NodeMCU/ESP8266 board [36]. A Raspberry Pi 3 B board is used as the central unit of the system, collecting data from all NodeMCU, and delivering information to the users. A similar approach was discussed in [37], which directly connected the ultrasonic sensors (in each parking spot) to a Raspberry Pi board, potentially decreasing the cost of the overall solution (but maybe turning it more complex to handle and more susceptible to failures). Moreover, the work in [37] considered the use of the Raspberry Camera to also detect the presence of cars.

For some solutions, the overall cost is a critical parameter and researchers struggle to develop a cheap system. In this sense and still addressing the problem of smart parking in urban areas, the work in [38] also employed ultrasonic sensors to detect the presence of cars, but instead of monitoring each parking spot individually, the detection of cars is performed only in the entrance and exit gates of each monitored parking area. Doing so, the accounting of the available parking spots is performed considering the total number of spots and the cars flow. In [38], a single Raspberry Pi board was used in each parking area as a central server, computing the available parking spots and delivering such information for the users. In fact, this approach is affordable when there are costs limitations, and it could be easily implemented in many cities.

Parking systems were also developed using the Arduino platform. In [39] a parking system employed Infra-Red (IR) sensors in each parking slot to detect if it is free. Then, an LCD display in the parking area entrance is used to exhibit information about the free spots. Moreover, RFID cards are used to support authorization management, since the parking area may not be public. All sensors and supporting hardware (as a motor to open the gate) are connected to an Arduino board, which makes all processing locally. Similarly, the work in [40] also monitored free parking spots using Arduino boards, which were connected to presence sensors (photosensitive). Additionally, an ESP8266 module was used to allow Wi-Fi communications by Arduino boards, enabling them to send information to an external system to inform users about the current parking system state. Differently, in [41] authors combined Arduino and Raspberry Pi platforms to support efficient parking, leaving the most computation burden to the Raspberry Pi board. As an interesting service, the work in [41] also employed CMOS sensors to detect the cars plates.

Following a different trend of the previously presented works, authors in [42] proposed a generic software solution for easy experimentation of visual sensor networks, which can be used in the context of smart cities. As a generic solution, the proposed approach could be used to perform many different functions. As an example of use, the work in [42] employed the BeagleBoard platform as a use case for parking management. The authors made an experiment using a BeagleBone Black board as a central unit, employing a camera to identify free parking spots. Although the work presented in [42] is generic in the sense that a complete software solution is proposed, the adoption of the BeagleBone Black as the open-source experimentation board also indicates its relevance for IoT applications.

3.2. Mobility

One of the main goals of smart-city systems is to enhance mobility, which can be achieved optimizing the public transportation system, the vehicles flows and the traffic conditions in rush hours. For that, different aspects can be addressed, and open-source electronics boards can be used to support solutions in this area.

A desired service for smart cities is vehicle tracking, since it is important for public transportation enhancement. The work in [43] proposed a bus tracking system, deploying a Raspberry board on each monitored bus. GPS (Global Positioning System) and GSM/GPRS (Global System for Mobile communications/General Packet Radio Service) expansion boards are attached to Raspberry Pi to provide localization information and data communication through the cellular network, respectively. Such services are then used to indicate the online location of the bus. A similar approach was performed in [44], but in that work a pre-determined route is defined for the vehicle trajectories. The idea was to send alert message if the driver picks a different route other than the one previously specified and already stored in the Raspberry Pi storage card. Moreover, auxiliary sensors for gas leakage and temperature were also employed, aiming at the monitoring of the safety conditions of the drivers.

In [45], the Arduino board was used to keep track of moving vehicles using GPS as the localization service. Then, a GSM/GPRS expansion board was employed to transmit positioning information to be displayed on any user smartphone. A similar work was conducted in [46], but now using a XBee expansion board (IEEE 802.15.4) instead of a GSM/GPRS module.

Concerning congestion and traffic jams, a relevant issue in smart cities is traffic-light management. The work in [47] employed a Raspberry Pi board to control the traffic lights to soften traffic jams in harsh hours. As traffic conditions may suddenly change, for example due to a car accident, the Raspberry Pi board could be used to adjust the time programming of traffic lights, but this service was not performed automatically in [47]. In that work, a user accesses the Raspberry board through an Android App and manually controls the traffic lights. From a different perspective, a Raspberry Pi board was used in [48] to measure the traffic flow (counting cars) and to automatically adjust the traffic lights programming, using a camera for that. The traffic light controllers can then transmit information among them using Wi-Fi connections, allowing a more efficient control of a set of traffic lights.

Still following this trend, the work in [49] employed Arduino boards to control traffic lights. The idea is to perform image processing to change the traffic lights programming, changing waiting times according to the detected vehicle flows on the lanes. In the proposed system, processing is performed in a “conventional” computer (webcams are connected to it) and only decisions are transmitted to two Arduino boards, which ultimately control the lights time. The Arduino platform was also used to support the Raspberry-based solution described in [48].

Also addressing the mobility issues in smart cities, but from a different perspective, the work in [50] leveraged RFID cards as a mechanism to manage passengers in a public system, controlled by a BeagleBone Black board. That board also controls advertisement and information displayed on LCD displays, integrating two important services expected from smart-city environments. From the same perspective, data gathered from every monitored bus is displayed on LCD panels in [51], employing a Raspberry Pi board to gather GPS information transmitted by buses (they have a Raspberry Pi board) and to display that information in a bus terminus.

3.3. Energy Management

Open-source boards can also be used to save energy and manage energy consumption in urban areas. The authors in [52] proposed a system to turn on lamp posts only in the presence of cars or pedestrian, instead of keeping the lamps always on during dark times. A Raspberry Pi board was used to control the entire system, being deployed as a single control node. Then, ZigBee communications [53] were performed to control the lamp posts on a defined area. The LEDs in the lamp posts are turned on when it is dark, and cars or pedestrians are detected through presence sensors. Those sensors are not connected to the Raspberry Pi board, but they communicate through Wi-Fi. A similar approach is presented in [54], but it used Arduino boards in all lamp posts, except the coordinator post, which was controlled by a Raspberry Pi board.

Still considering efficient management of lamp posts, the work in [55] employed an Arduino board to control the lights and to monitor the temperature and humidity of a region. As LED lamps may have their light intensity adjusted, authors proposed the use of LDR (Light Dependent Resistors) sensors to reduce light intensity (and energy consumption) when a region is already too bright, potentially achieving efficient lighting on an area. An Arduino board was used in [55] to integrate the sensor units and to perform the adjustments.

An interesting work was presented in [56]. In that work, a neural network was employed to reduce energy consumption in residences, mainly in peak hours, which may affect energy distribution planning in the neighborhoods. As energy consumption in a city is not uniform, and different fares may apply for different periods of the day, the proposed system acts reducing energy consumption when demand is naturally higher. The proposed solution is mainly processed in a Raspberry Pi board.

3.4. Discussions and Comments

There are many works proposing IoT and smart-city applications to solve relevant problems in urban areas. The presented and discussed works are only some examples of feasible projects that have open-source electronics boards as the processing core of the proposed solutions. Table 2 summarizes these smart-city prototypes and projects that are centered around at least one of the presented open-source development platforms.

Table 2. Some smart-city applications based on open-source electronics development platforms.

Work	Year	Model	Sensors	Smart-City Challenge
Vakula and Kolli [35]	2017	Raspberry Pi 3 B	Ultrasonic	Parking system
Ramaswamy [37]	2016	Raspberry Pi B	Ultrasonic, Camera	Parking system
Zadeh and Cruz [38]	2016	Raspberry Pi B+	Ultrasonic	Parking system
Chaudhary et al. [39]	2017	Arduino (model not specified)	IR	Parking system
Huang et al. [40]	2017	Arduino Nano	Presence	Parking system
Kanteti et al. [41]	2017	Raspberry Pi and Arduino (models not specified)	CMOS camera, Ultrasonic	Parking system
bondi et al. [42]	2015	BeagleBone Black	Camera	Parking system
Nalawade and Akshay [43]	2016	Raspberry Pi 3 B	GPS	Mobility
Shinde and Mane [44]	2015	Raspberry Pi B+	GPS, Gas, Temperature	Mobility
Rahman et al. [45]	2016	Arduino Uno	GPS	Mobility
Ibraheem and Hadi [46]	2018	Arduino Uno	GPS	Mobility
hariff et al. [50]	2016	BeagleBone Black	RFID	Mobility
Vakula and Raviteja [51]	2017	Raspberry Pi 3	GPS	Mobility
Misbahuddin et al. [47]	2015	Raspberry Pi B	-	Mobility
Basil and Sawant [48]	2017	Raspberry Pi 3, Arduino Uno	Camera	Mobility
Khushi [49]	2017	Arduino Uno	-	Mobility
Leccese et al. [52]	2014	Raspberry Pi B	-	Energy efficiency
Sunehra and Rajasri [54]	2017	Raspberry Pi 3 B, Arduino Uno	Presence, Ultrasonic	Energy efficiency
Dheena et al. [55]	2017	Arduino Nano	LDR, Temperature, Humidity	Energy efficiency
Mahapatra et al. [56]	2017	Raspberry Pi 3 B	Energy smart meter	Energy efficiency

As can be seen in Table 2, the surveyed works were published in recent years as expected. Although Arduino is a bit older, the Raspberry and the BeagleBoard platforms are still very young, but their potentials as development platforms for smart-city applications are continuously being proved. Their open-source nature, the reduced energy consumption and low acquisition costs are some of the characteristics that are making these boards winning choices.

4. Perspectives and Future Directions

The adoption of open-source electronics development platforms as key elements to construct monitoring and control applications has the potential to accelerate the expected transformations promoted by the IoT era [57], with smart cities, vehicular networks, smart farming, integrated health assistance and other disruptive applications potentially changing the way we perceive and interact

with the world. The projects and products being created with such open-source boards are becoming increasingly common and this trend will still be more evident with transformations in the educational system: children are having early contact with these boards, supporting the creation of a maker generation [58,59].

For the particular maturation of the smart-city initiatives, the presented open-source hardware platforms will continue to have an important role. In addition, there are some perspectives in this area that are relevant when envisioning future directions for them. Next subsections discuss some of these perspectives.

4.1. Multimedia Sensing

The development of affordable tiny camera and microphone hardware components to be connected to open-source boards opened new possibilities for monitoring applications. Different manufacturers are continuously releasing new extension boards and specialized electronics components, which has facilitated the designing of smart-city solutions based on multimedia data. The selection of the most appropriate multimedia components is then a relevant project choice.

For Raspberry Pi, the success of the original board encouraged the creation of a specialized camera that could be easily installed, being initially released in 2013. The Raspberry Camera is a low-power high-definition small camera that is shipped with a flexible flat cable to be plugged into the CSI (Camera Serial Interface). For the BeagleBoard and Arduino platforms, some specialized cameras have good performance and affordable prices, such as the CMOS OV7670. Concerning microphones, there are different components and the acquisition cost will be a function of the desired sensibility and audio capturing quality.

Besides choosing the proper multimedia electronic components, the development of smart-city applications based on image, video and audio will also have to be concerned with other issues, since its inherent higher complexity will directly impact storage, processing, and data transmission. Additionally, more complex tasks such as video compressing may also demand high processing power, which may prohibit the use of some resource-constrained boards. One of the trends for this problem is the adoption of development extension boards, as described in Section 4.2. Such extensions may bring additional processing power or even an extra GPU (Graphical Processing Unit), better supporting multimedia-based applications.

Some works have exploited multimedia sensing for IoT applications. The work in [60] investigated the transmission of high-definition video on the Raspberry Pi platform (particularly the Raspberry Pi 2 B), considering time constraints (real-time transmissions). The Raspberry Camera is used to transmit H.264 video streams and authors in [60] concluded that the desired high-definition real-time video transmissions were satisfactorily performed using the Raspberry Pi board and an IoT P2P communications framework. Considering the BeagleBone Black board, the work in [61] proposed a framework to create smart nodes that can handle any kind of data, including audio, image, and video, making easier the development of distributed sensor nodes. For the work in [62], audio sensors were created using a Raspberry Pi 2 B board. In that work, different audio processing tasks were performed in the Raspberry board, aiming at the calculation of different audio noise parameters. Audio processing for noise detection was also performed in [63], which created audio sensing nodes with Raspberry Pi 3 B boards. All these recent works are leveraging the processing capabilities of open-source electronics platforms for multimedia data processing and storage, which may bring significant results for smart-city scenarios.

4.2. Development Extension Boards

One of the perspectives when using open-source electronics platforms is the availability of development extension boards. Such extensions improve the capabilities of the boards, typically adding additional processing power, sensing units, or even facilitating prototyping. Therefore, the choosing

of development extension boards may be an important project decision and the constant releasing of new products reinforces the use of open-source boards for IoT and smart-city applications.

Development extension boards may be designed and produced by the same manufacturers of the open-source electronics platforms or even by other companies. Of course, this is possible due to the open-source nature of the platforms. In this way, this “extension boards market” is active and dynamical, constantly bringing novelties that can further support the development of significant applications for smart cities.

The connection of an extension board to an open-source board is performed exploiting the I/O pins. As most of these extensions will attach to the boards in a way that they will be over them, they are referred following this idea: for Raspberry Pi they are HAT (Hardware Attached on Top); for BeagleBoard, they are referred to as *capes*; and for Arduino, they are called *shields*.

Table 3 presents some popular extension boards for the surveyed open-source electronics platforms.

Table 3. Some popular extension boards for open-source electronics boards.

Extension Board	Platform	Resources	Manufacturer
Matrix Creator	Raspberry Pi	FPGA, LED array, Multiple sensors, Accelerometer, 8 microphones, ZigBee	http://www.matrix.one
Matrix Voice	Raspberry Pi	FPGA, LED array, 8 microphones, Wi-Fi/Bluetooth (ESP8266)	http://www.matrix.one
PiJuice HAT	Raspberry Pi	Portable battery	http://www.pijuice.com
Ethernet shield	Arduino	Ethernet communications capability	http://www.arduino.cc
BeagleBone proto cape	BeagleBone family	Prototyping support	http://www.beagleboard.org
GPS Logger Shield	Arduino	GPS	http://www.sparkfun.com
1Sheeld	Arduino	Integration of smartphone's resources with Arduino	http://1sheeld.com
SparkFun Pro RF	Arduino	LoRa communications capability	http://www.sparkfun.com

Among those extension boards, some of them can be very helpful for many smart-city applications. As an example, the Matrix Creator board has several onboard sensor units, which can provide many important sensed information for applications in an integrated way. Other interesting extension board is the 1Sheeld, which integrates smartphones and their sensors to Arduino (also providing software to facilitate this integration). Doing so, for example, the smartphone's camera and microphone can be used by the Arduino board, opening interesting opportunities. New extension boards are still being released and many of them may positively impact the development of IoT and smart-city applications.

Figure 3 presents an example of two development extension boards (HAT) for the Raspberry Pi platform.

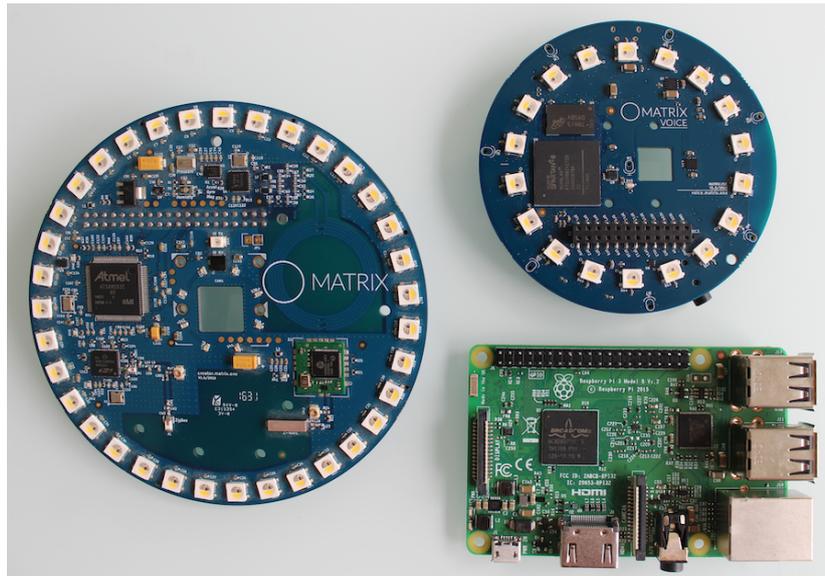


Figure 3. Examples of different development HAT for Raspberry Pi.

4.3. New Communication Technologies

The evolving of communications standards has a relevant role in the development of smart cities. Early applications were based on Wi-Fi technologies (more recently supported by ESP modules) but the advent of the IEEE 802.15.4 and ZigBee standard considerably changed this scenario. When extension boards supporting the ZigBee standard (mostly XBee modules) started to be released, a new wave of developments was possible [53,64]. In addition, we can expect such waves when new communication standards are released, once open-source boards are upgraded, and new extensions are released to allow such communications.

Concerning the perspectives in this area, a new communication standard can significantly enrich IoT smart-city applications. In a short term, a new standard, LoRa, has come as an efficient long-range low-power wireless communication standard that fits very well into the IoT demands [65]. Recently, some LoRa expansion boards were released, and new developments in this area are expected.

Other promising technology is 5G cellular networks. With this communication technology, high-bandwidth low-latency communication is possible, with higher communication range than current cellular networks. However, although promising, the introduction of 5G into the IoT world is still slow and other technologies are winning the race. With new releases of extensions boards, LoRa is being more believed to fulfill the IoT dreams and other standards such as Wi-Fi and Bluetooth are continuously evolving to increase bandwidth and reduce latency while keeping energy consumption low. The recent Raspberry Pi 3 B+ and Raspberry Pi 3 A+ models provide onboard support to the IEEE 802.11ac standard, allowing more efficient Wi-Fi communications. Some more years are still required to say what will be the winning technology for smart-city applications.

Therefore, multimedia sensing, real-time decisions and large data processing and storage will be some of the hot-topics to be addressed in smart-city applications and open-source electronics boards will play an important role to support the achievement of these goals. When concerning new communication technologies, the materialization of new standards in extensions boards and revised versions of open-source platforms will be crucial for the evolution of IoT and smart cities.

5. Conclusions

The advent of new hardware and communication technologies in the beginning of this century anticipated a series of innovations in the way information is gathered and processed, and initial developments of wireless sensor networks were first steps in this direction. However, although specialized hardware platforms were created to support sensing networks, such as the MicaZ [66] and

the TelosB [67], and they had an important role in the consolidation of sensor networks, the era of open-source electronics platforms is coming to enhance not only wireless sensor networks and their variations, but also to boost the complete IoT landscape. This article is then a valuable resource when trying to understand the transformations we are still living.

This article is an important contribution for new developments in this area, providing information that can guide new researches and projects for IoT applications. The discussions and surveyed works may contribute to create a vibrating atmosphere around open-source electronics platforms, which will help them to become the main development solution for smart cities. In addition, other emerging areas such as smart agriculture, smart retail, smart health, among others, may also benefit from those platforms.

The open-source electronics boards are still evolving, and new products are continuously being released. New platforms such as the NodeMCU [36,68] are still coming to the scene, opening new possibilities for small and big developers. In this context, we can expect a promising future for smart-city initiatives, when manufacturers and governmental agencies are seeing this as a fertile environment for the so eagerly awaited transformations.

Author Contributions: Authors contributed equally to the development of this work. D.G.C. and C.D.-F. contributed with the articles surveys and the discussions about the perspectives of the area. Both authors participated in the writing of the manuscript.

Funding: The APC was funded by the University of the Bío-Bío under grants DIUBB GI 160210/EF and DIUBB 184110 3/R.

Conflicts of Interest: The authors declare no conflict of interest. The founding sponsors had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, and in the decision to publish the results.

References

1. Lin, J.; Yu, W.; Zhang, N.; Yang, X.; Zhang, H.; Zhao, W. A Survey on Internet of Things: Architecture, Enabling Technologies, Security and Privacy, and Applications. *IEEE Internet Things J.* **2017**, *4*, 1125–1142. [[CrossRef](#)]
2. Talari, S.; Shafie-khah, M.; Siano, P.; Loia, V.; Tommasetti, A.; Catalão, J.P.S. A Review of Smart Cities Based on the Internet of Things Concept. *Energies* **2017**, *10*, 421. [[CrossRef](#)]
3. Peixoto, J.P.J.; Costa, D.G. Wireless visual sensor networks for smart city applications: A relevance-based approach for multiple sinks mobility. *Future Gener. Comput. Syst.* **2017**, *76*, 51–62. [[CrossRef](#)]
4. Zanella, A.; Bui, N.; Castellani, A.; Vangelista, L.; Zorzi, M. Internet of Things for Smart Cities. *IEEE Internet Things J.* **2014**, *1*, 22–32. [[CrossRef](#)]
5. Costa, D.G.; Collotta, M.; Pau, G.; Duran-Faundez, C. A Fuzzy-Based Approach for Sensing, Coding and Transmission Configuration of Visual Sensors in Smart City Applications. *Sensors* **2017**, *17*, 93. [[CrossRef](#)] [[PubMed](#)]
6. Glasmeier, A.K.; Nebiolo, M. Thinking about Smart Cities: The Travels of a Policy Idea that Promises a Great Deal, but So Far Has Delivered Modest Results. *Sustainability* **2016**, *8*, 1122. [[CrossRef](#)]
7. Allam, Z.; Newman, P. Redefining the Smart City: Culture, Metabolism and Governance. *Smart Cities* **2018**, *1*, 4–25. [[CrossRef](#)]
8. Molina, B.; Palau, C.E.; Fortino, G.; Guerrieri, A.; Savaglio, C. Empowering smart cities through interoperable Sensor Network Enablers. In Proceedings of the 2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC), San Diego, CA, USA, 5–8 October 2014; pp. 7–12.
9. Fortino, G.; Savaglio, C.; Zhou, M. Toward opportunistic services for the industrial Internet of Things. In Proceedings of the 2017 13th IEEE Conference on Automation Science and Engineering (CASE), Xi'an, China, 20–23 August 2017; pp. 825–830.
10. Costa, D.G.; Duran-Faundez, C.; Andrade, D.C.; Rocha-Junior, J.B.; Just Peixoto, J.P. TwitterSensing: An Event-Based Approach for Wireless Sensor Networks Optimization Exploiting Social Media in Smart City Applications. *Sensors* **2018**, *18*, 1080. [[CrossRef](#)]

11. Santana, E.F.Z.; Chaves, A.P.; Gerosa, M.A.; Kon, F.; Milojevic, D.S. Software Platforms for Smart Cities: Concepts, Requirements, Challenges, and a Unified Reference Architecture. *ACM Comput. Surv.* **2017**, *50*, 1–37. [[CrossRef](#)]
12. Fortino, G.; Savaglio, C.; Palau, C.E.; de Puga, J.S.; Ganzha, M.; Paprzycki, M.; Montesinos, M.; Liotta, A.; Llop, M., Towards Multi-layer Interoperability of Heterogeneous IoT Platforms: The INTER-IoT Approach. In *Integration, Interconnection, and Interoperability of IoT Systems*; Gravina, R., Palau, C.E., Manso, M., Liotta, A., Fortino, G., Eds.; Springer International Publishing: Cham, Switzerland, 2018; pp. 199–232.
13. Costa, D.G.; Guedes, L.A. Exploiting the sensing relevancies of source nodes for optimizations in visual sensor networks. *Multimedia Tools Appl.* **2013**, *64*, 549–579. [[CrossRef](#)]
14. Tavade, T.; Nasikkar, P. Raspberry Pi: Data logging IOT device. In Proceedings of the 2017 International Conference on Power and Embedded Drive Control (ICPEDC), Chennai, India, 16–18 March 2017; pp. 275–279.
15. Kruger, C.P.; Hancke, G.P. Benchmarking Internet of things devices. In Proceedings of the 2014 12th IEEE International Conference on Industrial Informatics (INDIN), Porto Alegre, Brazil, 27–30 July 2014; pp. 611–616.
16. Nikhade, S.G. Wireless sensor network system using Raspberry Pi and zigbee for environmental monitoring applications. In Proceedings of the Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM), Chennai, India, 6–8 May 2015; pp. 376–381.
17. Patil, N.; Ambatkar, S.; Kakde, S. IoT based smart surveillance security system using raspberry Pi. In Proceedings of the Communication and Signal Processing (ICCSP), Chennai, India, 6–8 April 2017; pp. 0344–0348.
18. Costa, D.G. On the Development of Visual Sensors with Raspberry Pi. In Proceedings of the 24th Brazilian Symposium on Multimedia and the Web, WebMedia '18, Salvador, Brazil, 16–19 October 2018; ACM: New York, NY, USA, 2018; pp. 19–22.
19. Patchava, V.; Kandala, H.B.; Babu, P.R. A Smart Home Automation technique with Raspberry Pi using IoT. In Proceedings of the Smart Sensors and Systems (IC-SSS), Bangalore, India, 21–23 December 2015; pp. 1–4.
20. He, Q.; Segee, B.; Weaver, V. Raspberry Pi 2 B+ GPU Power, Performance, and Energy Implications. In Proceedings of the 2016 International Conference on Computational Science and Computational Intelligence (CSCI), Las Vegas, NV, USA, 15–17 December 2016; pp. 163–167.
21. Paramanathan, A.; Pahlevani, P.; Thorsteinsson, S.; Hundeboll, M.; Lucani, D.E.; Fitzek, F.H.P. Sharing the Pi: Testbed Description and Performance Evaluation of Network Coding on the Raspberry Pi. In Proceedings of the 2014 IEEE 79th Vehicular Technology Conference (VTC Spring), Seoul, Korea, 18–21 May 2014; pp. 1–5.
22. Pedersen, M.V.; Heide, J.; Fitzek, F.H.P. Kodo: An Open and Research Oriented Network Coding Library. In *NETWORKING 2011 Workshops*; Casares-Giner, V., Manzoni, P., Pont, A., Eds.; Springer: Berlin/Heidelberg, Germany, 2011; pp. 145–152.
23. Nunes, L.H.; Nakamura, L.H.V.; de F. Vieira, H.; de O. Libardi, R.M.; de Oliveira, E.M.; Estrella, J.C.; Reiff-Marganiec, S. Performance and energy evaluation of RESTful web services in Raspberry Pi. In Proceedings of the 2014 IEEE 33rd International Performance Computing and Communications Conference (IPCCC), Austin, TX, USA, 5–7 December 2014; pp. 1–9.
24. Maduranga, M.W.P.; Ragel, R.G. Comparison of load balancing methods for Raspberry-Pi Clustered Embedded Web Servers. In Proceedings of the 2016 International Computer Science and Engineering Conference (ICSEC), Chiang Mai, Thailand, 14–17 December 2016; pp. 1–4.
25. Nayyar, A.; Puri, V. A Review of Beaglebone Smart Board's-A Linux/Android Powered Low Cost Development Platform Based on ARM Technology. In Proceedings of the 2015 9th International Conference on Future Generation Communication and Networking (FGCN), Jeju, Korea, 25–28 November 2015; pp. 55–63.
26. Samie, F.; Bauer, L.; Henkel, J. IoT technologies for embedded computing: A survey. In Proceedings of the 2016 International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS), Pittsburgh, PA, USA, 2–7 October 2016; pp. 1–10.
27. Heo, Y.J.; Oh, S.M.; Chin, W.S.; Jang, J.W. A Lightweight Platform Implementation for Internet of Things. In Proceedings of the 2015 3rd International Conference on Future Internet of Things and Cloud, Rome, Italy, 24–26 August 2015; pp. 526–531.

28. Hejazi, H.; Rajab, H.; Cinkler, T.; Lengyel, L. Survey of platforms for massive IoT. In Proceedings of the 2018 IEEE International Conference on Future IoT Technologies (Future IoT), Eger, Hungary, 18–19 January 2018; pp. 1–8.
29. Sicari, S.; Rizzardi, A.; Miorandi, D.; Coen-Porisini, A. Securing the smart home: A real case study. *Internet Technol. Lett.* **2018**, *1*, e22. [[CrossRef](#)]
30. Costantino, D.; Malagnini, G.; Carrera, F.; Rizzardi, A.; Boccadoro, P.; Sicari, S.; Grieco, L.A. Solving Interoperability within the Smart Building: A Real Test-Bed. In Proceedings of the 2018 IEEE International Conference on Communications Workshops (ICC Workshops), Kansas City, MO, USA, 20–24 May 2018; pp. 1–6.
31. Gunpath, S.; Murdan, A.P.; Oree, V. Design and implementation of a low-cost Arduino-based smart home system. In Proceedings of the 2017 IEEE 9th International Conference on Communication Software and Networks (ICCSN), Guangzhou, China, 6–8 May 2017; pp. 1491–1495.
32. Pradhan, M.A.; Patankar, S.; Shinde, A.; Shivarkar, V.; Phadatare, P. IoT for smart city: Improvising smart environment. In Proceedings of the 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS), Chennai, India, 1–2 August 2017; pp. 2003–2006.
33. Vidiasova, L.; Kachurina, P.; Cronemberger, F. Smart Cities Prospects from the Results of the World Practice Expert Benchmarking. *Procedia Comput. Sci.* **2017**, *119*, 269–277. [[CrossRef](#)]
34. Al Nuaimi, E.; Al Neyadi, H.; Mohamed, N.; Al-Jaroodi, J. Applications of big data to smart cities. *J. Internet Serv. Appl.* **2015**, *6*, 25. [[CrossRef](#)]
35. Vakula, D.; Kolli, Y.K. Low cost smart parking system for smart cities. In Proceedings of the 2017 International Conference on Intelligent Sustainable Systems (ICISS), Palladam, India, 7–8 December 2017; pp. 280–284.
36. Barai, S.; Biswas, D.; Sau, B. Estimate distance measurement using NodeMCU ESP8266 based on RSSI technique. In Proceedings of the 2017 IEEE Conference on Antenna Measurements Applications (CAMA), Tsukuba, Japan, 4–6 December 2017; pp. 170–173.
37. Ramaswamy, P. IoT smart parking system for reducing green house gas emission. In Proceedings of the 2016 International Conference on Recent Trends in Information Technology (ICRITIT), Chennai, India, 8–9 April 2016; pp. 1–6.
38. Zadeh, N.R.N.; Cruz, J.C.D. Smart urban parking detection system. In Proceedings of the 2016 6th IEEE International Conference on Control System, Computing and Engineering (ICCSCE), Batu Ferringhi, Malaysia, 25–27 November 2016; pp. 370–373.
39. Chaudhary, H.; Bansal, P.; Valarmathi, B. Advanced CAR parking system using Arduino. In Proceedings of the 2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 6–7 January 2017; pp. 1–5.
40. Huang, K.Y.; Chang, S.B.; Tsai, P.R. The advantage of the arduino sensing system on parking guidance information systems. In Proceedings of the 2017 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), Singapore, 10–13 December 2017; pp. 2078–2082.
41. Kanteti, D.; Srikar, D.V.S.; Ramesh, T.K. Intelligent smart parking algorithm. In Proceedings of the 2017 International Conference On Smart Technologies For Smart Nation (SmartTechCon), Bangalore, India, 17–19 August 2017; pp. 1018–1022.
42. Bondi, L.; Baroffio, L.; Cesana, M.; Redondi, A.; Tagliasacchi, M. EZ-VSN: An Open-Source and Flexible Framework for Visual Sensor Networks. *IEEE Internet Things J.* **2016**, *3*, 767–778. [[CrossRef](#)]
43. Nalawade, S.R.; Akshay, S.D. Bus tracking by computing cell tower information on Raspberry Pi. In Proceedings of the 2016 International Conference on Global Trends in Signal Processing, Information Computing and Communication (ICGTSPICC), Jalgaon, India, 22–24 December 2016; pp. 87–90.
44. Shinde, P.A.; Mane, Y.B. Advanced vehicle monitoring and tracking system based on Raspberry Pi. In Proceedings of the 2015 IEEE 9th International Conference on Intelligent Systems and Control (ISCO), Coimbatore, India, 9–10 January 2015; pp. 1–6.
45. Rahman, M.M.; Mou, J.R.; Tara, K.; Sarkar, M.I. Real time Google map and Arduino based vehicle tracking system. In Proceedings of the 2016 2nd International Conference on Electrical, Computer Telecommunication Engineering (ICECTE), Rajshahi, Bangladesh, 8–10 December 2016; pp. 1–4.
46. Ibraheem, I.K.; Hadi, S.W. Design and Implementation of a Low-Cost Secure Vehicle Tracking System. In Proceedings of the 2018 International Conference on Engineering Technology and their Applications (IICETA), Al-Najaf, Iraq, 8–9 May 2018; pp. 146–150.

47. Misbahuddin, S.; Zubairi, J.A.; Saggaf, A.; Basuni, J.; A-Wadany, S.; Al-Sofi, A. IoT based dynamic road traffic management for smart cities. In Proceedings of the 2015 12th International Conference on High-capacity Optical Networks and Enabling/Emerging Technologies (HONET), Islamabad, Pakistan, 21–23 December 2015; pp. 1–5.
48. Basil, E.; Sawant, S.D. IoT based traffic light control system using Raspberry Pi. In Proceedings of the 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS), Chennai, India, 1–2 August 2017; pp. 1078–1081.
49. Khushi. Smart Control of Traffic Light System using Image Processing. In Proceedings of the 2017 International Conference on Current Trends in Computer, Electrical, Electronics and Communication (CTCEEC), Mysore, India, 8–9 September 2017; pp. 99–103.
50. Shariff, S.U.; Swamy, J.C.N.; Seshachalam, D. Beaglebone black based e-system and advertisement revenue hike scheme for Bangalore city public transportation system. In Proceedings of the 2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology (iCATcT), Bangalore, India, 21–23 July 2016; pp. 781–786.
51. Vakula, D.; Raviteja, B. Smart public transport for smart cities. In Proceedings of the 2017 International Conference on Intelligent Sustainable Systems (ICISS), Palladam, India, 7–8 December 2017; pp. 805–810.
52. Leccese, F.; Cagnetti, M.; Trinca, D. A Smart City Application: A Fully Controlled Street Lighting Isle Based on Raspberry-Pi Card, a ZigBee Sensor Network and WiMAX. *Sensors* **2014**, *14*, 24408–24424. [[CrossRef](#)]
53. Wang, H.; Dong, L.; Wei, W.; Zhao, W.; Xu, K.; Wang, G. The WSN Monitoring System for Large Outdoor Advertising Boards Based on ZigBee and MEMS Sensor. *IEEE Sensors J.* **2018**, *18*, 1314–1323. [[CrossRef](#)]
54. Sunehra, D.; Rajasri, S. Automatic street light control system using wireless sensor networks. In Proceedings of the 2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI), Chennai, India, 21–22 September 2017; pp. 2915–2919.
55. Dheena, P.P.F.; Raj, G.S.; Dutt, G.; Jinny, S.V. IOT based smart street light management system. In Proceedings of the 2017 IEEE International Conference on Circuits and Systems (ICCS), Thiruvananthapuram, Indian, 20–21 December 2017; pp. 368–371.
56. Mahapatra, C.; Moharana, A.K.; Leung, V.C.M. Energy Management in Smart Cities Based on Internet of Things: Peak Demand Reduction and Energy Savings. *Sensors* **2017**, *17*, 2812. [[CrossRef](#)]
57. Kurkovsky, S.; Williams, C. Raspberry Pi as a Platform for the Internet of Things Projects: Experiences and Lessons. In Proceedings of the 2017 ACM Conference on Innovation and Technology in Computer Science Education, ITiCSE '17, Bologna, Italy, 3–5 July 2017; pp. 64–69.
58. Adams, J.C.; Brown, R.A.; Kawash, J.; Matthews, S.J.; Shoop, E. Leveraging the Raspberry Pi for CS Education. In Proceedings of the 49th ACM Technical Symposium on Computer Science Education, SIGCSE '18, Baltimore, MD, USA, 21–24 February 2018; pp. 814–815.
59. Pattichis, M.S.; Celedon-Pattichis, S.; LopezLeiva, C. Teaching image and video processing using middle-school mathematics and the Raspberry Pi. In Proceedings of the 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, LA, USA, 5–9 March 2017; pp. 6349–6353.
60. Jennehag, U.; Forsstrom, S.; Fiordigigli, F.V. Low Delay Video Streaming on the Internet of Things Using Raspberry Pi. *Electronics* **2016**, *5*, 60. [[CrossRef](#)]
61. Chianese, A.; Piccialli, F.; Riccio, G. Designing a Smart Multisensor Framework Based on Beaglebone Black Board. In *Computer Science and Its Applications*; Park, J.J.H., Stojmenovic, I., Jeong, H.Y., Yi, G., Eds.; Springer: Berlin/Heidelberg, Germany, 2015; pp. 391–397.
62. Noriega-Linares, J.E.; Navarro Ruiz, J.M. On the Application of the Raspberry Pi as an Advanced Acoustic Sensor Network for Noise Monitoring. *Electronics* **2016**, *5*, 74. [[CrossRef](#)]
63. Peckens, C.; Porter, C.; Rink, T. Wireless Sensor Networks for Long-Term Monitoring of Urban Noise. *Sensors* **2018**, *18*, 3161. [[CrossRef](#)]
64. De Oliveira, K.V.; Castelli, H.M.E.; Montebeller, S.J.; Avancini, T.G.P. Wireless Sensor Network for Smart Agriculture using ZigBee Protocol. In Proceedings of the 2017 IEEE First Summer School on Smart Cities (S3C), Natal, Brazil, 6–11 August 2017; pp. 61–66.
65. Pasolini, G.; Buratti, C.; Feltrin, L.; Zabini, F.; De Castro, C.; Verdone, R.; Andrisano, O. Smart City Pilot Projects Using LoRa and IEEE802.15.4 Technologies. *Sensors* **2018**, *18*, 1118. [[CrossRef](#)]

66. Ali, N.A.; Drieberg, M.; Sebastian, P. Deployment of MICAz mote for Wireless Sensor Network applications. In Proceedings of the 2011 IEEE International Conference on Computer Applications and Industrial Electronics (ICCAIE), Penang, Malaysia, 4–7 December 2011; pp. 303–308.
67. Guevara, J.; Vargas, E.; Brunetti, F.; Barrero, F. Open architecture for WSN based on runtime reconfigurable systems and the IEEE 1451. In Proceedings of the 2013 IEEE SENSORS, Baltimore, MD, USA, 3–6 November 2013; pp. 1–4.
68. Araújo, A.; Kalebe, R.; Girão, G.; Filho, I.; Gonçalves, K.; Melo, A.; Neto, B. IoT-Based Smart Parking for Smart Cities. In Proceedings of the 2017 IEEE First Summer School on Smart Cities (S3C), Natal, Brazil, 6–11 August 2017; pp. 31–36.



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).