Creating a sensor tier for the EMULSION IoT platform with low-cost electronic modules

PAPER • OPEN ACCESS

Creating a Sensor Tier for the EMULSION IoT Platform with Low-Cost Electronic Modules

To cite this article: Ivan Ganchev and Zhanlin Ji 2022 *J. Phys.: Conf. Ser.* **2226** 012009

Creating a Sensor Tier for the EMULSION IoT Platform with Low-Cost Electronic Modules

Ivan Ganchev[1,2,3*] **and Zhanlin Ji**[3,4]

[1] Department of Computer Systems, University of Plovdiv "Paisii Hilendarski", Plovdiv, Bulgaria
[2] Institute of Mathematics and Informatics – Bulgarian Academy of Sciences, Sofia, Bulgaria
[3] Telecommunications Research Centre (TRC), University of Limerick, Limerick, Ireland
[4] College of Artificial Intelligence, North China University of Science and Technology, P. R. China

[*]Ivan.Ganchev@ul.ie

Abstract. This paper presents some of the designed and experimentally-tested low-cost electronic modules, utilized for the creation of a sensor tier for the generic, multi-service, cloud-based operational platform EMULSION, which is being elaborated for rapid building of mobile Internet of Things (IoT) systems and roll-out of corresponding IoT services. The next step is to achieve full integration of the designed modules with various other existing components with different processing and communication capabilities, as to allow 1,000,000 heterogeneous IoT nodes, deployed at the sensor tier of the platform, to communicate simultaneously online within a single EMULSION cluster.

1. Introduction

Becoming the main driving force of the future global economic development, the Internet of Things (IoT) [1] [2] is seen as another information and industrial wave following those of personal computers, the Internet, and mobile communication networks.

IoT finds application in multiple domains. For example, sensors backed up by mobile IoT devices are installed on bridges to monitor their vibration amplitude and predict possible collapse. Similar devices could be installed in forests to generate early warnings about possible fires. The hillsides in hot and humid areas may have IoT devices, installed to monitor the ground displacement. Mobile IoT devices could be installed also on river sides to monitor the rainfall or on farmlands to monitor plant diseases and insect pests, etc.

For the provision of such services, relevant IoT systems are required running on a suitable IoT platform, allowing a transparent communication with different types of IoT nodes and offering (value-added) functionalities, e.g., remote device control and management, telco/dew/fog/cloud and storage management, 'big data' analytics and visualization, application enablement, etc., with some sort of freedom of use by (and customization to) users [3], which in the cloud domain is known as PaaS (Platform as a Service). A multi-service, cloud-based IoT operational platform EMULSION, which could serve as a generic architectural foundation for a 'smart city' development [4], has been proposed in [5]. EMULSION is able to support heterogeneous-type communication and data exchange between different types of IoT applications and nodes. Our goal is to develop this platform for future rapid roll-out of IoT systems and services of different types, whereby a single cluster can support up to 1,000,000 IoT nodes.

Designing a robust and reliable, but at the same low-cost, IoT platform, encompassing a variety of hardware (e.g., semiconductors, processors, single-board computers, power-supply and communication modules, etc.) and software (e.g., embedded operating systems, distributed message queue producer-consumer subsystems, databases, machine learning APIs, cloud(s), applications, tools, utilities, etc.) is a difficult task. Even with millions of repositories of hardware information and open-source software available on the GitHub, choosing the right electronic modules and software code in numerous technical routes is still a significant challenge both for researchers and developers.

EMULSION is being developed by means of low-cost electronic components (and open-source software), based on the generic IoT system architecture depicted in figure 1. Using this architecture, the sensors (S), location trackers (T) [6], and monitoring stations (MS), deployed at the sensor tier, can communicate with the corresponding data/remote transfer unit (D/RTU), equipped with relevant communication modules (e.g., Bluetooth, Wi-Fi, LoRa, 2G÷6G, etc.) to reach –via a wireless sensor network, if needed, and a smart communication gateway– the corresponding information center or server in the cloud tier, which collects and analyzes the aggregated data from all D/RTUs in order to provide relevant services to users via the service and client tiers or obtain respective commands/decisions/recommendations that are sent back to the actuators (A), controllers (C), and guards (G) in the sensor tier.

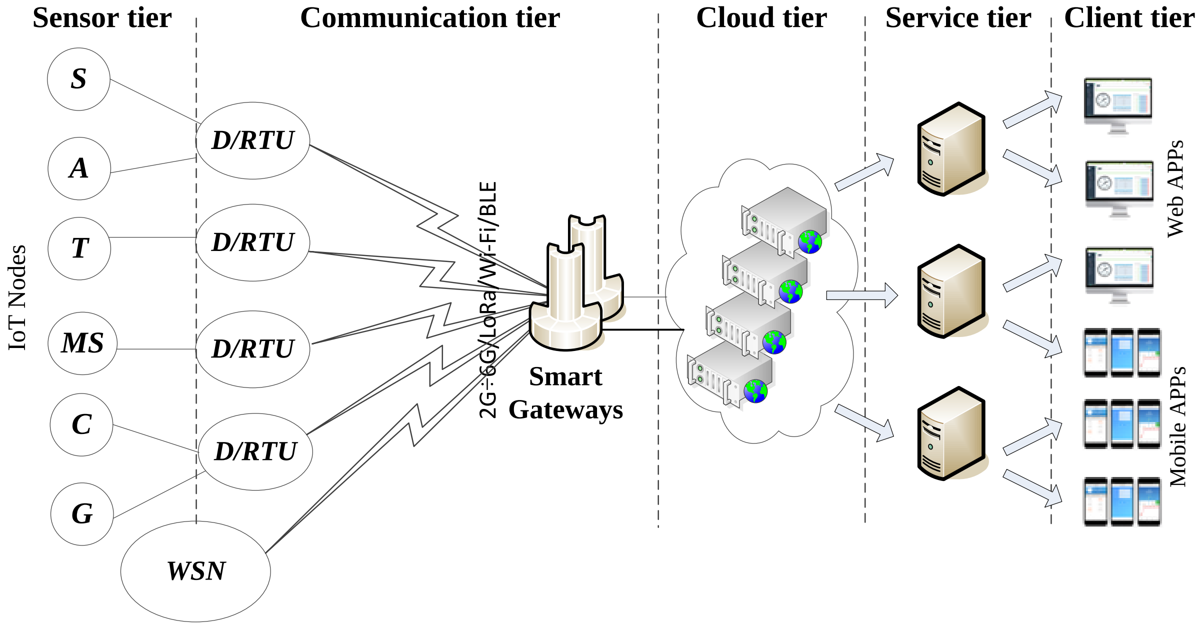


Figure 1. A generic IoT system architecture.

The rest of the paper is structured as follows. Next, the EMULSION platform is briefly presented, followed by a description of methods used for the creation of its sensor tier. The presentation of the designed low-cost electronic modules comes next, followed by a brief discussion and conclusion.

2. EMULSION

Our aim is to develop EMULSION as a representative of the next-generation, *horizontal* type, IoT platforms replacing the existing *vertical*-type ones, which are specialized to provide services within a single IoT domain only. The result of applying the vertical approach was the emergence of the so-called 'Internet of Silos' (instead of real 'Internet of Things'), causing fragmentation and closed infrastructural boundaries, whereby each IoT provider delivers entire vertical, involving separate services, applications, network connections, things, objects, nodes, etc., and focusing on a particular

IoT domain, e.g., energy/smart grid, industry, supply chain, smart cities/buildings/homes, smart environment protection, smart healthcare, smart agriculture, etc. This way by the end of 2019, there were 620 IoT platform vendors presented in the market, with the top 10 of them having 58% market share [3]. This vertical trend led to even more IoT market fragmentation with accompanying problems related to system interoperability and integration, increased OPEX, limited openness to new services and use cases, and difficult scaling.

Following the horizontal approach, we are developing EMULSION as an open IoT platform, primarily responding to the needs of SMEs. By utilizing the integration and interoperability principles, this horizontal trend allows to simplify the existing IoT environment by eliminating duplicate solutions and enables inter-technology operation and generation of new IoT services and business opportunities. Equipped with this new solution, a service/application/network provider can deliver a complete horizontal-slice, applied to not just one but multiple IoT domains, meeting by this the flexibility-, cost-efficiency-, multi-purpose use, and scalability and lifecycle requirements (e.g., for supporting millions of IoT devices distributed over 10+ years, [7]) with efficient control and management of the entire IoT ecosystem throughout its lifetime.

Our goal with EMULSION is to provide a generic horizontal-type IoT platform solution for use with any type of IoT node, device, cloud, domain, service, etc. For this, the platform was conceived having a multi-tiered, service-oriented, flexible and easily expandable, cloud-based structure [5].

3. Methods

The methods we apply in the development of EMULSION are focused on the utilization of low-cost electronic modules and open-source software components.

In [8], a low-cost electrical smart power meter prototype was proposed, based on a STM32F2 microcontroller unit (MCU), but its price and hardware design were not clearly stated. In [9], SEnviro IoT devices were proposed for smart farming, utilizing a MCU board costing €63.41. In [10], a low-cost Wi-Fi based IoT prototype was proposed, costing around US $60 for a single unit. In [11], a low-cost IoT system for irrigation monitoring and control, utilizing ESP-12F 8266 as MCU, was proposed, with a cost of US $54.90. In [12], low-cost agriculture devices were built with Raspberry Pi 3. However, with a typical cost of around US $40, Raspberry Pi 3 seems still expensive, if compared, for instance, with the open-source hardware boards running Linux and Android, whose popularity has increased recently. A comparison of different (partially) open-source hardware MCU boards with 2G/3G connectivity is presented in [9]. Compared to these works, our goal was to design GPRS/NB-IoT/LTE/Wi-Fi core boards, with a single cost of less than US $10, for use in various mobile IoT systems.

In addition to hardware, the choice of software also plays an important role in reducing the overall cost of an IoT platform. Most IoT platforms today run as commercial software products, some transitioned from academic research, e.g., [13]. The commercial IoT platforms, however, often promote vendor lock-in [14]. One of the emerging trends is the openness of IoT platforms due to major cost savings brought along with the convenient and fast development. There are different platform openness types, e.g., related to the open-source, open standards, open APIs, open data, open layers, etc. Among these, the open standards and open-source play a significant role especially for IoT system integrators, who provide end-to-end solutions by integrating different parts (e.g., sensors, actuators, controllers, communication modules, business logic software components, applications, etc.), as to decrease the switching cost of creating a better IoT ecosystem [14]. The complex nature of IoT platforms, encompassing different servers, databases (DBs), applications, message queues, interface, etc., makes quite challenging the selection of suitable open-source components that can interoperate successfully on one hand and meet the IoT hardware requirements on the other. Just taking the DB selection as an example, even though the performance of some DBs, e.g., MongoDB vs. MySQL [15], in different cloud environments has been well studied, for quite many IoT applications, Redis and InfluxDB [16] are the most preferred DB types.

The use of a successful combination of the designed low-cost electronic modules with the created open-source software components, based on Nginx [17], Kafka [18], Netty, Redis [19], InfluxDB, and TDEngine [20], was demonstrated while building the EMULSION platform.

In EMULSION, a Nginx web server is used, acting as a concurrent reverse proxy to redirect HTTP streams to internal servers in a high-performance fashion. This way, supported by Nginx modules, the platform can work in a distributed mode, without stopping its operation even in the case of servers' crashes by utilizing, for instance, the following stream's *upstream stream_backend* configuration:

```
stream {
    upstream stream_backend {
        server backend1.ucww.com:10081 weight=5;
        server backend2.ucww.com:10082;
        server backend3.ucww.com:10083; max_conns=4000;
    }
}
```

The IoT data streams passed through Nginx are delivered to a distributed publish-subscribe messaging Kafka cluster with defined rules for decoupling the input generators from processors [18]. The multi-broker Kafka cluster is managed by a leader, elected among the brokers, which is also responsible for creating the input and output partitioned topics on which the IoT data streams are made available (figure 2). Each producer publishes (with a constant or burst rate depending on the needs) its own data sets in the form of messages on the corresponding Kafka input topic. Each broker, responsible for partitions of the topic logs, can handle up to several hundreds of thousands of messages per second. With customized low-level implementations, the processing flows can be greatly optimized, and messages can be sent out to consumers as soon as possible with low delays [18]. Consumers read messages (a subset of the topic partitions) from the topics they have subscribed to. The Apache ZooKeeper is used by brokers for managing and coordinating the cluster. For instance, when brokers and topics are added/removed, ZooKeeper informs all nodes in the cluster so they can coordinate their actions and elect new partition leaders, if required.

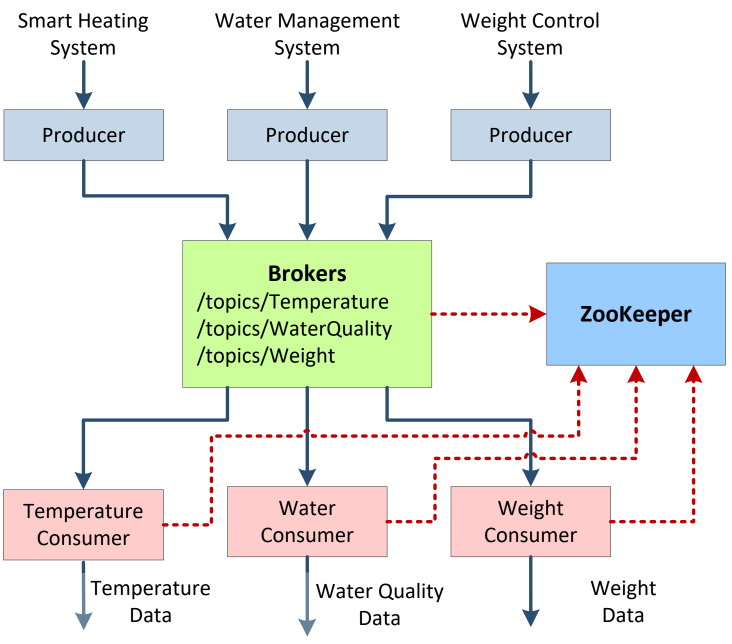


Figure 2. The publish-subscribe Kafka messaging configuration used.

Different from information systems used in other areas, in IoT, usually one device owns one table. There are several time series databases suitable for use in IoT systems. In our R&D work, we selected InfluxDB for use as a single-server database and TDEngine as a distributed database. In addition, we use MySQL for saving station information, Hadoop HBase for history data, and Redis [19] for in-memory caching, communication sessions' maintaining, etc.

The Spring Boot and Spring Cloud are two widely used tools in distributed systems for quickly building common patterns, communicating with multiple services, and collecting data in one application. In an IoT system with 100,000 devices, for instance, more than 10 applications may need to work together to provide IoT services, such as login, alarming, caching, configuring, managing, monitoring, pushing, messaging, etc. Figure 3 depicts the common Spring Boot application package structure, utilized in our IoT R&D work.

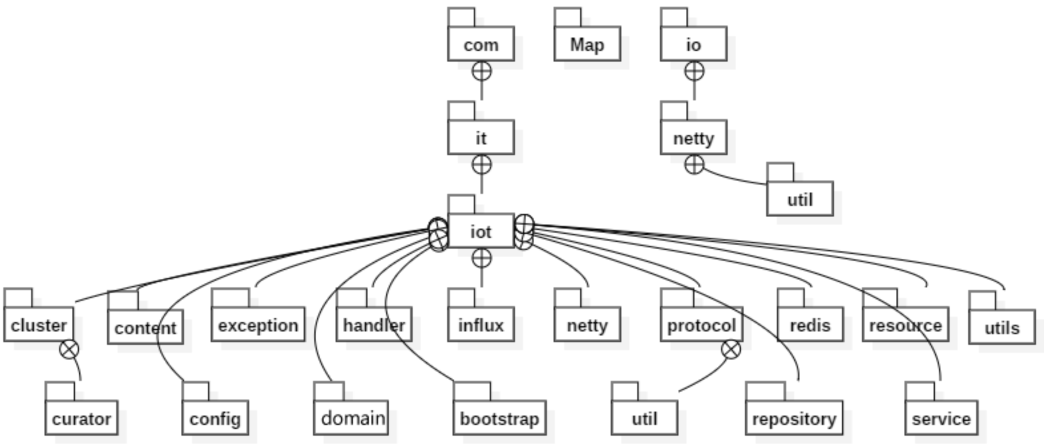


Figure 3. A common Spring Boot application package structure.

4. Low-Cost Electronic Modules

This paper is presents only low-cost electronic solutions. Expensive high-speed solutions, involving 1000+ MHz MCUs and multi-core configurations, are not considered here.

4.1. Power-Supply Modules

These modules are important for guaranteeing the stand-alone functioning of battery-powered IoT devices. Consecutively, their design is crucial for the successful operation of the entire IoT platform. For example, an LTE communication module needs a power supply with at least 2A current for its normal operation. The corresponding power-supply module designed includes a AC-to-DC converter supporting ‘220V/110V to 12V’ conversion, a DC-to-DC converter, a ‘12V to 5V’ switch, and a ‘5V to 3.3V/1.8V’ low dropout (LDO) voltage regulator. Figure 4 depicts the circuit diagram of the designed DC-to-DC converter.

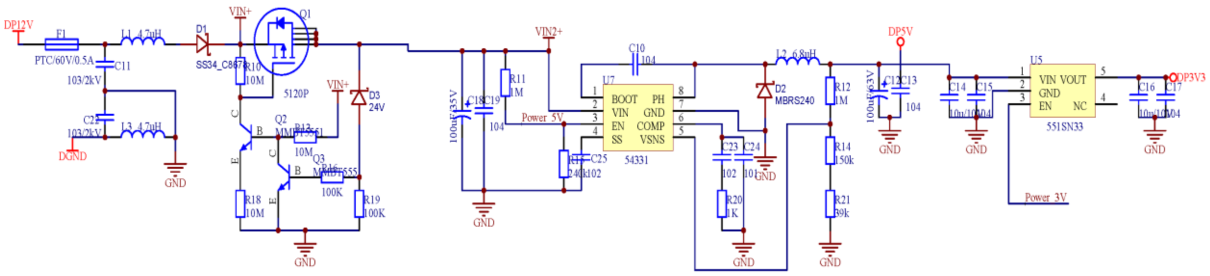


Figure 4. The circuit diagram of a designed DC-to-DC converter.

In this converter, the key element is the TPS54331 chip, providing a 5V/3A output. For micropower applications, an additional LDO voltage regulator is usually required, e.g., TPS70933DBVR supporting a 12V input and a 3.3V output, and a quiescent current of only 1μA that enables the IoT hardware to work on batteries for many years.

4.2. MCU Modules

The MCU design, allowing remote update and upgrade, is very important as this is a key requisite for any IoT platform. In most of our IoT endeavors, we used the Arm Cortex-M3 STM32F103RE to design single-chip systems. It has a 512-kB flash memory, a 72-MHz CPU, and can work in a temperature range of –40°C to +105°C. To communicate with sensors, it provides three 12-bit ADCs, two I2Cs, 11 timers, three SPIs, and five USARTs. To enable rapid development, we have designed the core board with Winbond W25Q32JVTCIQ flash, and 8M and 32M crystal oscillators. The layout of the developed printed circuit board (PCB) is shown in figure 5.

Several real-time operating systems (RTOS) can support this MCU module, for example, µC/OS-II, FreeRTOS, RTX, Huawei LiteOS, RT-Thread, etc. The latter was selected for utilization by EMULSION because, besides providing a RTOS kernel, it also supports a great number of drivers for IoT peripherals and sensors. Moreover, more than 100 IoT software packages were released to support the file system, database, communication, debugging, MicroPython scripting language, etc. The utilized software architecture is depicted in figure 6.

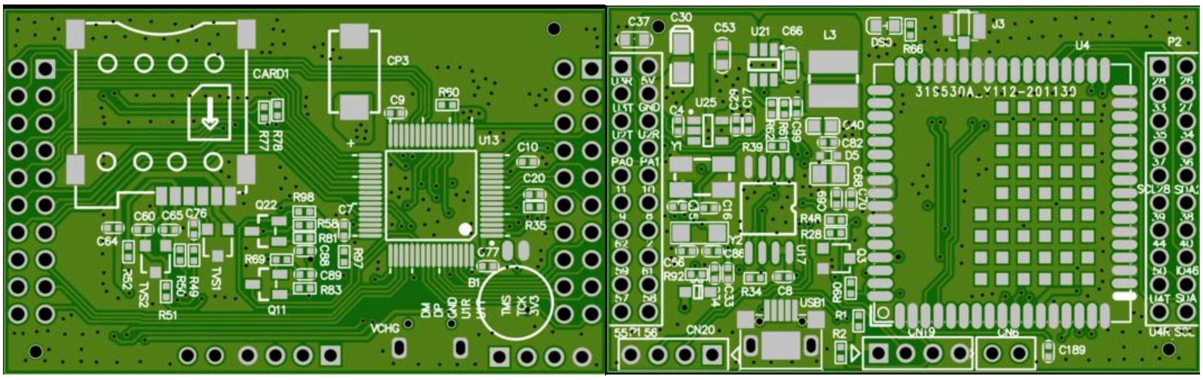


Figure 5. The PCB layout of a designed MCU module's core board.

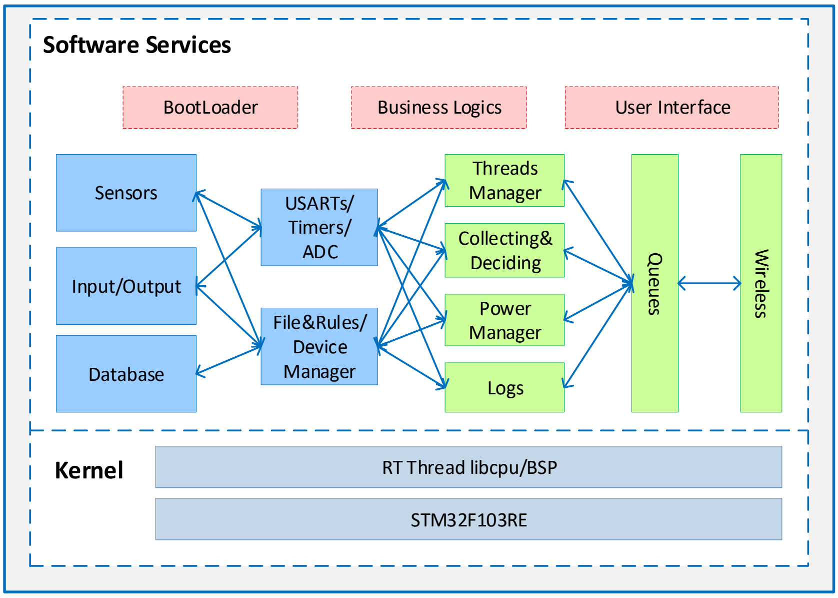


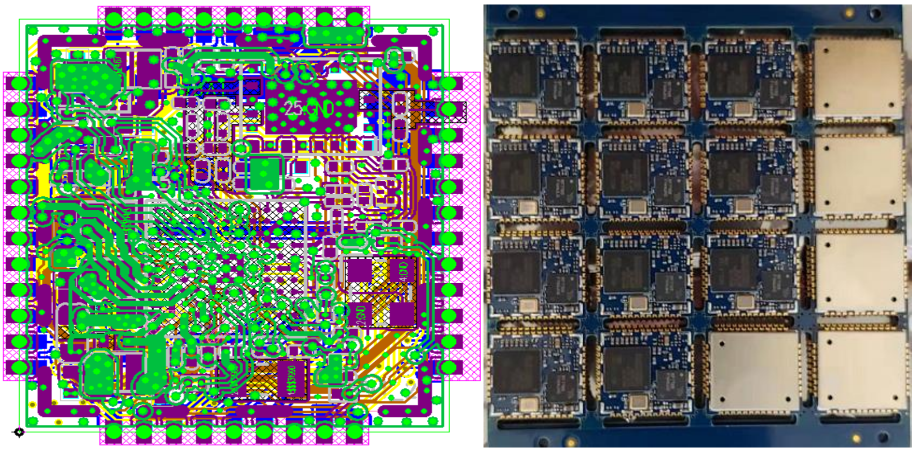
Figure 6. The software architecture for the MCU module's core board.

4.3. Communication Modules

These are the key components in any IoT platform. Four types of wireless communication modules have been developed: (1) GPRS (80%); (2) Wi-Fi (10%); (3) NB-IoT (5%); and (4) LTE Cat.1/Cat.4 modules (5%). The GPRS modules utilize MTK 2503 chipsets, the NB-IoT modules use MTK 2625 chipsets, the Wi-Fi modules are based on ESP32 chipsets, and the LTE Cat.1 and Cat.4 modules utilize UIS8910DM and MTK 6737 chipsets, respectively.

To date, we primarily focused on the GPRS (2G) type of communication modules as the cheapest implementation option available nowadays. The Bill of Materials (BOM), including the PCB cost, of a single GPRS module is around 11 RMB (1.4 Euro), whereas for NB-IoT and LTE modules it comes to 30 RMB (3.8 Euro) and 70 RMB (9 Euro), respectively. Wi-Fi was our least popular option, simply because changing the password of a Wi-Fi router by its owner may cause 'disappearance' of all IoT devices accessible through it. NB-IoT, while being an attractive option, suffers from a signal-coverage problem, for instance, in the countryside of China.

The PCB of the developed MTK2503-based GPRS module (figure 7a) was designed with 8 layers. It uses a buried/blind via hole (BVH) technique and fully considers the characteristic impedance. figure 7b depicts the developed modules on a joint board. Table 1 contains the corresponding BOM.



(a) (b)

Figure 7. The developed MTK2503-based GPRS modules': (a) PCB layout; (b) joint board.

Table 1. The BOM of the developed MTK2503-based GPRS module.

Component	Quantity	Notation	Specification
RF Crystal	1	X100	Crystal 3225 26MHz 7.5pF
LEDs	2	DS1, DS2	0805 LED
INDs	1	L100	2.7nH±0.3nH, L0201
	4	L102, L103, L104, L105	4.3nH±0.3nH, L020
ICs	2	U1, U2	IC MT2503 MCU
	1	U3	IC RF7198 RF PA
Capacitors	30	C121, C122, etc.	C0201 28 C0603 2
Resistors	18	R1, R2, etc.	100R 330K

5. Discussion

Due to the 'know-how' nature of the low-cost electronic solutions presented in this paper for the creation of mobile IoT systems, further technical details cannot be included here.

Along with the presented electronic solutions, novel algorithms, techniques, models, and open-source software solutions are being elaborated for the effective provision of IoT services based on the EMULSION platform, primarily focusing on smart healthcare and smart environment protection but also with a capacity for application in other IoT domains. The aim is to ensure the delivery of the best QoE to users when using different IoT services, accessible through any type of mobile device and wireless access network, *anytime-anywhere-anyhow*. EMULSION can supply highly personalized, customized, and contextualized IoT services by taking into account the user-, service-, and (access) network context. By utilizing distributed 'big data' management techniques and an efficient real-time cloud computing based environment, at the end, EMULSION will be able to turn the real-world raw sensing data and the collected user behavior in using IoT services into rich analytic datasets as to proactively recommend the best service instances applicable to each individual user under the ABC&S communications paradigm [21].

6. Conclusion

This paper has presented some of the designed low-cost electronic modules (i.e., MCUs, communication modules, and power-supply modules), required for building a sensor tier for the IoT operational platform EMULSION, which is being elaborated to support heterogeneous-type communication and data exchange between various types of IoT devices and applications, with the required flexibility, multidimensional scalability, efficiency, interoperability, and easy adjustment to new IoT scenarios and use cases.

Acknowledgments

This publication has emanated from joint research conducted with the financial support of the Bulgarian National Science Fund (BNSF) under the Grant No. КП-06-ИП-КИТАЙ/1 (KP-06-IP-CHINA/1) and the S&T Major Project of the Science and Technology Ministry of China under the Grant No. 2017YFE0135700.

References

1. Pujolle, G. *An Autonomic-oriented Architecture for the Internet of Things*. in *IEEE John Vincent Atanasoff 2006 International Symposium on Modern Computing (JVA'06)*. 2006.
2. Atzori, L., A. Iera, and G. Morabito, *The Internet of Things: A survey.* Computer Networks, 2010. **54**(15): p. 2787-2805.
3. Lueth, K.L., *IoT Platform Companies Landscape 2019/2020: 620 IoT Platforms globally*. 2019, IoT Analytics.
4. Ganchev, I., Z. Ji, and M. O'Droma. *A generic IoT architecture for smart cities*. in *25th IET Irish Signals & Systems Conference 2014 and 2014 China-Ireland International Conference on Information and Communications Technologies (ISSC 2014/CIICT 2014)*.
5. Ganchev, I., Z. Ji, and M. O'Droma, *Designing a Cloud Tier for the IoT Platform EMULSION.* WSEAS Transactions on Systems and Control, 2019. **14**: p. 375-383.
6. Ganchev, I., Z. Ji, and M. O'Droma, *Designing a Low-Cost Location Tracker for Use in IoT Applications*, in *XXXIII General Assembly and Scientific Symposium of the International Union of Radio Science (URSI GASS 2020)*. 2020, IEEE. p. x1-x2.
7. HP, *Delivering on the IoT customer experience: The HPE Universal IoT Platform 1.4*. 2020. p. 1-24.
8. Abate, F., et al., *A low cost smart power meter for IoT.* Measurement, 2019. **136**: p. 59-66.
9. Trilles, S., A. González-Pérez, and J. Huerta, *A Comprehensive IoT Node Proposal Using Open Hardware. A Smart Farming Use Case to Monitor Vineyards.* Electronics, 2018. **7**(12): p. 419.
10. Ciuffoletti, A., *Low-Cost IoT: A Holistic Approach.* Journal of Sensor and Actuator Networks, 2018. **7**(2): p. 19.
11. Borah, S., R. Kumar, and S. Mukherjee, *Low-cost IoT framework for irrigation monitoring and control.* International Journal of Intelligent Unmanned Systems, 2021. **9**(1): p. 63-79.
12. Lavanya, G., C. Rani, and P. Ganeshkumar, *An automated low cost IoT based Fertilizer Intimation System for smart agriculture.* Sustainable Computing: Informatics and Systems, 2020. **28**: p. 100300.
13. Cirillo, F., et al., *A Standard-Based Open Source IoT Platform: FIWARE.* IEEE Internet of Things Magazine, 2019. **2**(3): p. 12-18.

14. Vogel, B., et al., *What Is an Open IoT Platform? Insights from a Systematic Mapping Study.* Future Internet, 2020. **12**(4): p. 73.
15. Eyada, M.M., et al., *Performance Evaluation of IoT Data Management Using MongoDB Versus MySQL Databases in Different Cloud Environments.* IEEE Access, 2020. **8**: p. 110656-110668.
16. Ahmad, K. and M. Ansari, *Hands-on InfluxDB*, in *NoSQL: Database for Storage and Retrieval of Data in Cloud*. 2017, Chapman and Hall/CRC. p. 341-354.
17. Butler, T., *NGINX Cookbook*. 2017: Packt Publishing Ltd.
18. van Dongen, G. and D. van den Poel, *Evaluation of Stream Processing Frameworks.* IEEE Transactions on Parallel and Distributed Systems, 2020. **31**(8): p. 1845-1858.
19. Ji, Z., et al. *A Distributed Redis Framework for Use in the UCWW*. in *2014 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery*. 2014.
20. TAOS. *TDengine Documentation*. 2021 17/01/2021]; Available from: https://www.taosdata.com/en/documentation/.
21. Ganchev, I., *User-Centric and Context-Aware ABC&S.* International Journal "Information Technologies & Knowledge", 2015. **9**(3): p. 203-215.