A Short Course on IoT

Ioannis Chatzigiannakis[1] and Andrea Vitaletti[1]

[1]Sapienza Università di Roma, Dipartimento di Ingegneria Informatica, Automatica e Gestionale, Rome, Italy; {ichatz,vitaletti}@diag.uniroma1.it

Abstract

We presents a short course on the Internet of Things designed to give a first understanding of the main issues in developing IoT solutions. The course is organized into four main sessions of 3 hours each. First session introduce the main IoT concepts and focus on the relationship of IoT with Big Data and on the constrained introduced by the limited availability of resources on IoT devices. The second session is focused on hands on. To simplify the experiments we exploit a convenient online simulator and Arduino IDE, focusing on the maker approach, namely in doing the right thing with minimal consideration in terms of efficiency. The third session focuses on the engineering approach where also efficiency is crucial and introduce a more advanced developing tool based on FreeRTOS, a Real Time Operating System for embedded devices. The last session introduce IoT-LAB an infrastructure to run experiments on multiple nodes. The last section of the paper discusses the lessons learned in the first edition of the course and introduce possible future work. All the material for the course is available at [1].

1 Introduction

This paper presents a short course on the Internet of Things designed to give a first understanding of the main issues in developing IoT solutions. The course is organized into four main sessions of 3 hours each.

The first session provides (see section 2) an introduction to the Internet of Things. In particular, we investigate the connection that exists between the IoT and Big Data. In our vision, IoT is the fuel of Big Data allowing to quantitatively measure phenomena that up to yesterday we were only able to guess. Thanks to such measures we can build models to better understand the observed phenomena and to predict future behaviors. In this way, we build actionable intelligence that allows us to consciously act in the environment. However, is crucial to understand that IoT means a network of resource-constrained devices: if computational power, memory, energy, bandwidth, and costs are not constrained, it is likely the Internet, not the Internet of Things. Such constraints need the introduction of concepts such as edge-computing and duty-cycles.

We do believe the most efficient way to learn this subject is by experimenting, consequently, the course encourages attendants to put hands-on. No specific background is needed, even if basic electronics and computer science knowledge might help. The second session (see section 3) is designed to start experimenting with real devices. The main goal is to allow students to integrate sensors suitable for their monitoring purposes. In this session, we pursue the Maker approach, namely we focus on effectiveness, i.e., doing the right things, without considering efficiency. The hands-on start on the very nice and practical simulator WOKWI [12]. It leverages the Arduino Programming language and it allows us to experiment with several hardware platforms and integrates a variety of sensors and actuators in the simplest, but realistic way. Indeed, after some simulations, students are ready to upload their projects into real devices. In our case, we used the well-known ESP32 platform. We exploited the ESP32-DevKit [5] which provides WiFI and BLE connectivity. During the course, we stress that even if such technologies are not well suited for the IoT, they represent a very practical means of experimenting with the IoT. Furthermore, Heltec provides an ESP32 device supporting LORA [7].

Once students have familiarised themselves with simple projects, it is time to clarify that the Maker approach is not sufficient. Session 3 (see section 4) is designed to introduce the Engineering approach focused on efficiency, namely doing things right. The main concept here is that we need metrics to measure to what extent our solution satisfies the user requirements. An interactive approach that step after step converges toward a satisfactory solution is encouraged. Only clear metrics allow us to measure to what extent we indeed converge. Furthermore, while Arduino programming language is a very nice introduction, we discuss the need for a more sophisticated RTOS supporting multitasking, Real Time Applications and Real Time Scheduling. To this purpose, in this section students experiment with FreeRTOS [6] running on ESP32.

Finally, session 4 (see section 5) is designed to scale-up the solutions developed in the previous sessions in a testbed supporting a multitude of nodes. To this purpose, we introduce the IoT-LAB [9], a very large-scale infrastructure suitable for testing small wireless sensor devices and heterogeneous communicating objects. Furthermore, IoT-LAB allows to measure the use of resources on the nodes in a very convenient way, so that students can easily understand the impact os simple duty-cycle approaches.

All the material for the course is available at [1]. The design of the course leverages the experience gained in teaching IoT in the MSC in Computer Engineering at Sapienza University of Rome [8].

The paper is concluded with section 6 in which we discuss the lessons learned during the first edition of the course and we introduce possible future work.

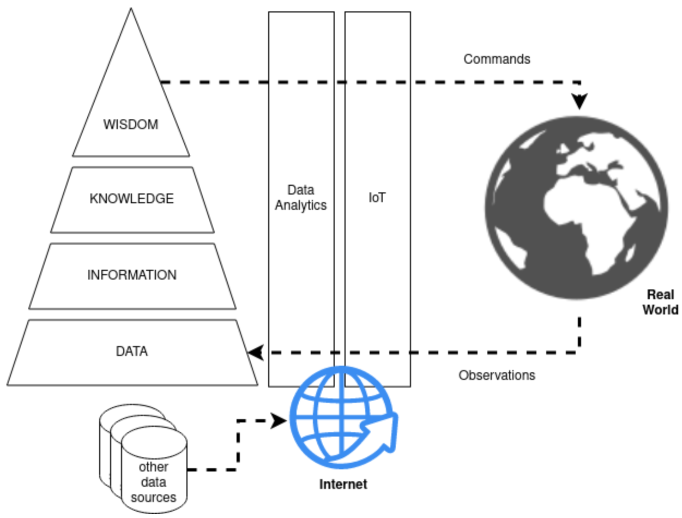

Figure 1: The IoT allows us to observe real world phenomena at an unprecedented level of granularity and to consciously act into he environment in view of informed decision obtained analysing the collected data

2 Introduction to the Internet of Things

The course is designed to be an introduction to the IoT. Most people, relegates IoT to a convenient way to implement remote control by mobile phones. This is obviously a nice feature of the IoT, but in our opinion IoT should be placed in a wider perspective. Lord Kelvin, the father of the first and second laws of thermodynamics, said: "When you can measure what you are speaking about, and express it in numbers, you know something about it". The IoT, is indeed a tool to measure phenomena at an unprecedented level of granularity that before the advent of such technology were observed only qualitatively. In other words, the IoT if fuel for Big Data Analytic [13]. This is summarised in figure 1. The IoT is made of resource constrained devices connected to the Internet. The Internet connectivity guarantee that IoT devices can be anywhere and can be interrogated at any time. Ubiquitous presence of IoT devices, allow us to collect data an unprecedented level of granularity on a multitude of phenomena. Such data, possibly enriched by other data available on the Internet, are distilled into actionable intelligence, thanks to data analytic tools such as Machine Learning. The actionable intelligence operates into the environment commanding the actuators of the IoT devices. The effectiveness of such actions is confirmed by new monitoring data in an iterative process trying to converge to optimal solutions.

The main goal of the first session is to illustrate this vision, however it is extremely important that students understand since the beginning that the IoT is a network of resource constrained devices. Nowadays, students deals with relatively powerful computers, providing GHz of clock, GB of ram, Mbps of connectivity and high capacity batteries that can be recharged every day. In contrast, a typical IoT device can have MHz of clock, hundreds of KB of RAM, few Kbps of connectivity (e.g. LORA) and a battery with limited capacity that in principle cannot be easily recharged. Moreover, the goal of ubiquitous deployments requires cost effective devices that limit the quality of hardware components.

These constraints impose a careful design of the IoT solutions [14] and the employment of energy aware approaches (e.g. duty-cycle) as well as to delegate the computation to more powerful devices on the edge when possible, just to mention two possible approaches. Furthermore, to support access to low-level functionalities, IoT coding is mostly done in C or C++ languages, and in some cases students need some time to familiarize with such languages. All in all, the design of reliable IoT solution requires to move from the maker approach, mostly focused on effectiveness, to an engineering approach where efficiency is a key design ingredient.

At the end of this session, as an exercise, students are encouraged to explore IoT projects presented in some well-known platforms (e.g. hackernoon) identifying possible issues in terms of the inefficient use of the resources.

3 First experiences with real hardware: the maker approach

The main goal of this session is to put hands on. We do believe this allows students to better understand the concepts discussed so far and to have fun in carrying out practical activities. Initially we pursue the maker approach, namely we focus on doing the right thing with minor concerns in terms of efficiency. To really simplify this activity, we initially focus on WOKWI [12] an online tool capable to realistically simulate many popular boards, parts and sensors. In particular we simulate an ESP32 programmed in the Arduino Programming language (see figure 2) performing the following four main activities that will be replicated also in the other sessions:

- The simplest actuator, namely a led
- The simplest digital sensor, namely a button
- An analog sensor, namely a potentiometer connected to the ADC
- A simple example with SR04 Ultrasonic Sensor.
- MQTT connectivity.

For the sake of convenience these activities rely on WiFi connectivity, but it is very important to stress since the beginning that this is a convenient option that however is inappropriate in many IoT deployments mostly due to its high energy requirements.

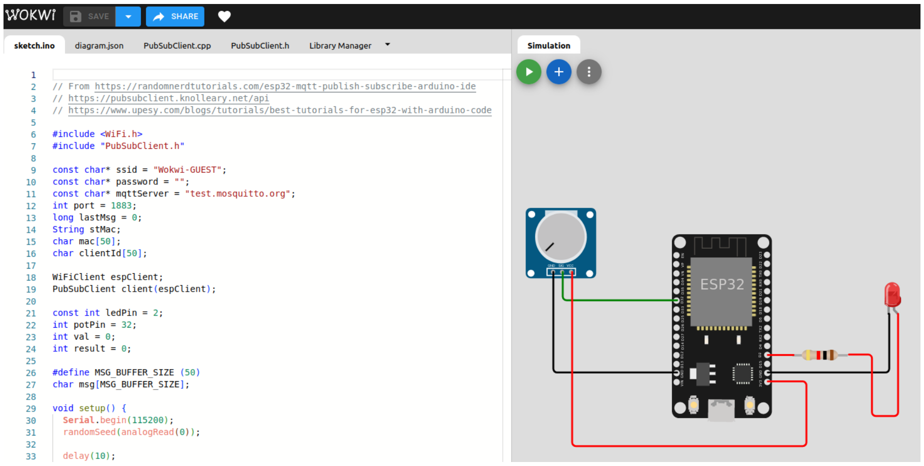

Figure 2: A relatively complex simulation where the data acquired by the ADC connected to the variable resistor are delivered to a back-end via MQTT. Similarly commands can be delivered to he code to switch on/off the led.

Once students are familiar with the above simulations, they are ready to experiment with real hardware. In particular we exploit the ESP32-DevKit [5], a very flexible ESP32 platform which provides WiFi and BLE connectivity. Heltec [7] provides a similar hardware also supporting LORA connectivity.

Running the experiments on real hardware (see figure 3) is straightforward since WOKWI allow us to download the Arduino code corresponding to the simulations that can be conveniently uploaded to the ESP32 using the Arduino IDE [4].

A simple back-end relying on thingsboard [11] allows us to collect the data by MQTT and visualise them in a nice dashboard as well as to send MQTT commands to the ESP32 via a nice interface.

4 The engineering approach: measure performance and exploit a RTOS

This session is focused on two main goals: a) use metrics to measure to what extent your solution satisfies the user requirements, b) introduce a real time operating system (RTOS) more suitable for reliable IoT deployments.

For the sake of simplicity we focus on a simple project based on [2], a paper presenting the results of the accident study findings on household uses of gas. We quickly analyse the document in the class to identify a set of minimal requirements to detect fires in the proximity of gas cabinets and to promptly inform the emergency center. Here we focus on a single requirement to clarify

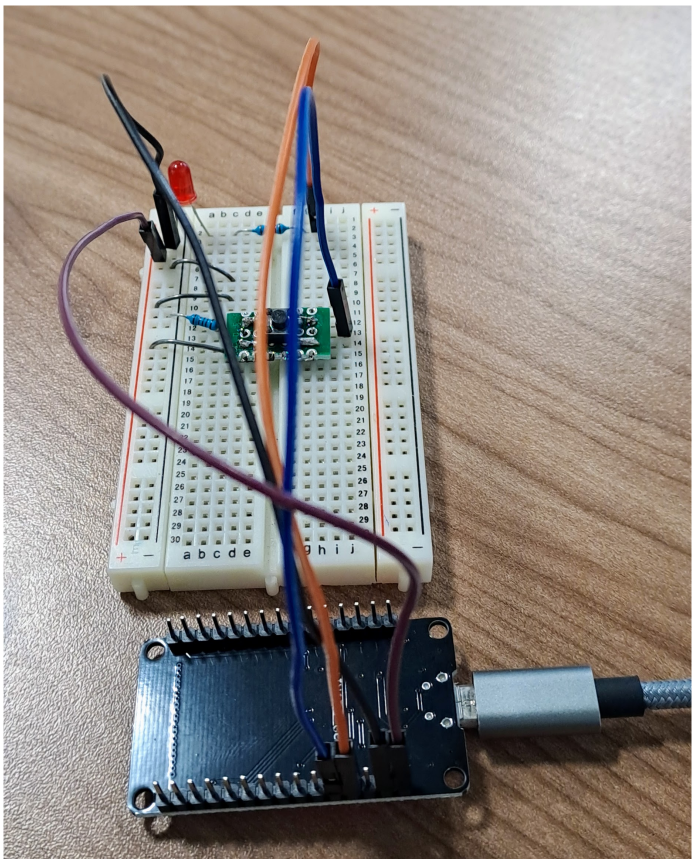

Figure 3: An ESP32 connected to a breadboard with a led and a button. The USB cable provides power and debug functionalities through the serial terminal

the purpose of this activity, namely the need of detecting a fire at 5 meters from the cabinet with an accuracy of at least 75% in at most 1 minute. We first analyse different sensors to detect fires and then we design an experiment in which sensor are placed at different distance from the fire source to measure their ability to reliably detect the episode. Students experience with concept such as accuracy, false positive and negatives, dependence of measures from relevant factors such as the distance, the presence of the sun etc. An iterative process to improve the original solution is encouraged as well as the design of experiments to quantify the quality of the solutions in view of the user requirements.

In the second part of the session we introduce the need of a Real-Time Operating Systems (RTOSs). IoT is usually characterised by real time requirements, namely IoT devices have to respond to a certain event within a strictly defined deadline. In the above example, we have to detect the fire in at most 1 minute. This requires a deterministic behaviour of the scheduler. In this session we introduce FreeRTOS [6], a state of the art RTOS, that supports the ESP32. In FreeRTOS the scheduler exploits user's defined priorities to know which thread of execution to run next. The ability of splitting a program in task with different priorities implies a dramatically different approach to coding if compared to the single loop of Arduino coding (see figure4).

```
//Arduino code

void setup() {
  // initialize digital pin
  // LED_GREEN and LED_RED
  // as an output.
}

void loop() {
  digitalWrite(LED_GREEN, HIGH);
  digitalWrite(LED_RED HIGH);
  delay(1000);
  digitalWrite(LED_GREEN, LOW);
  digitalWrite(LED_RED, LOW);
  delay(1000);
}
```

```
//FreeRTOS code

void led_red_blink(void *pvParams) {
    gpio_pad_select_gpio(LED_RED);
    gpio_set_direction (LED_RED,
                GPIO_MODE_OUTPUT);
    while (1) {
        gpio_set_level(LED_RED,0);
        vTaskDelay(1000/portTICK_RATE_MS);
        gpio_set_level(LED_RED,1);
        vTaskDelay(1000/portTICK_RATE_MS);
    }
}
// same structure for function led_green_blink
void app_main() {
    xTaskCreate(&led_red_blink,"LED_BLINK",
            512,NULL,5,NULL);
    xTaskCreate(&led_green_blink,"LED_BLINK",
            512,NULL,5,NULL);
}
```

Figure 4: A simple example to blink two leds. On the left the Arduino code based on a single loop. On the right the the FreeRTOS code wth multiple tasks.

Students analyse how to split the program in different independent tasks that can be coordinated when needed by mechanisms exploiting suitable the OS functionalities. This activity is concluded porting the activities encoded in session 2 (see section 3) in Arduino into FreeRTOS. A rich set of examples is available at `https://github.com/espressif/esp-idf`

5 Leverage existing IoT infrastructures

Up to this section, students have experimented with a single device with different tools at different level of complexity. They proved to be able to integrate simple sensors and actuators on real hardware and to communicate acquired data to a back-end as well to get commands from it. However, real IoT deployments are usually made of multiple devices. To explore scalability issues we leverage on IoT-LAB [9]. It is a very large scale infrastructure suitable for testing small wireless sensor devices and heterogeneous communicating objects. Remarkably, IoT-LAB supports FreeRTOS code `https://iot-lab.github.io/docs/os/freertos/`, even if the ESP32 is not currently among the supported hardware. This proves to students the capability of FreeRTOS to run on heterogeneous hardware platforms the same code. IoT-LAB allows us to experimenting with ad-hoc networking and LORA, a communication technology more suitable for the IoT, interacting with the The Thing Network [10] a global collaborative Internet of Things ecosystem that creates networks, devices and solutions using LoRaWAN. Finally, it also allow us to collect fine grained data on the use of resources [3], bandwidth, power consumption etc, thus supporting the engineering approach introduced in the previous session.

6 Discussion and Future Work

The first edition of the course has been given in August 2003 at the University of Stellenbosch, Sudafrica, during and Eramus+ teaching mobility exchange. Forty students among more than an hundred of candidates have been selected to participate. Students provided positive feedback to the course and really appreciated the hands on (see the post on Linkedin https://acesse.one/bAvKP).

Sessions 1,2 and 3 have been successfully delivered, while session 4 has been only introduced. In other words, 12 hours to deliver all the sessions is over optimistic, realistically all the sessions can be delivered in at least 16 hours.

All the material is publicly available on github [1]. The choice of providing a virtualbox appliance with the development environment and all the necessary toolboxes already deployed and configured has paid off. Likely, the most time consuming activity in view of the heterogeneous OSs used by students (mostly Windows and MacOS in different flavour) , has been set-up the SSH connection with the appliance and the interoperability with a local IDE (e.g. vscode). To further simplify this issue, the current version of the virtual appliance provides micro, a convenient text editor that can run on an SSH terminal.

This course is our attempt to summarise a minimal set of topics to introduce students to the IoT in view of the gained in teaching IoT in the MSC in Computer Engineering at Sapienza University of Rome [8]

A number of relevant topics such as security have not been considered with the goal of design a simple and practical course to be delivered in relatively short time. However, supplementary material is already available on interesting issues, such as Kalman filters and Machine Learning on embedded devices. Depending on the length of the course and the background of the students, many extensions can be considered, however, we consider a deep understanding of the concepts introduced in session 1,2 and 3 crucial to deal with IoT beyond the maker approach.

The material of the course is always work in progress and we really hope to improve it in view of the feedback provided by attendees, colleagues and experts. Collaborations are very welcome.

Acknowledgement

This work is partially supported by the Master Degree in Industry 4.0 610455-EPP-1-2019-1-My-EPPKA2-CBHE-JP (https://www.ind4-0-eu.my/index.php). The material has been also developed during an ERASMUS+ ICM_TEACHING MOBILITY program of Andrea Vitaletti to the University of Stellenbosch in Sudafrica.

References

[1] A Short Course on IoT . https://andreavitaletti.github.io/IoT_short_course/ [Online; accessed Sept. 2023].

[2] Accident study findings on household uses of gas. https://www.aria.developpement-durable.gouv.fr/wp-content/uploads/2016/03/2016-03-01_SY_utilgaz_JFM_GB_Vfin.pdf [Online; accessed Sept. 2023].

[3] Consumption monitoring in IoT-LAB. https://iot-lab.github.io/docs/tools/consumption-monitoring/ [Online; accessed Sept. 2023].

[4] Download and install Arduino IDE . https://support.arduino.cc/hc/en-us/articles/360019833020-Download-and-install-Arduino-IDE [Online; accessed Sept. 2023].

[5] ESP32-DevKitC V4 Getting Started Guide. https://docs.espressif.com/projects/esp-idf/en/latest/esp32/hw-reference/esp32/get-started-devkitc.html [Online; accessed Sept. 2023].

[6] FreeRTOS™ Real-time operating system for microcontrollers . https://www.freertos.org/ [Online; accessed Sept. 2023].

[7] Heltec ESP32 Node series. https://docs.heltec.org/en/node/esp32/index.html [Online; accessed Sept. 2023].

[8] INTERNET OF THINGS: ALGORITHMS AND SERVICES . http://ichatz.me/Site/InternetOfThings2023 [Online; accessed Sept. 2023].

[9] IoT-LAB The Very Large Scale IoT Testbed . https://www.iot-lab.info/ [Online; accessed Sept. 2023].

[10] The Thing Network. https://www.thethingsnetwork.org/ [Online; accessed Sept. 2023].

[11] ThingsBoard: Open-source IoT Platform. Device management, data collection, processing and visualization for your IoT solution. https://thingsboard.io/ [Online; accessed Sept. 2023].

[12] WOKWI - Simulate IoT Projects in Your Browser. https://wokwi.com/ [Online; accessed Sept. 2023].

[13] Y. Nait Malek, A. Kharbouch, H. El Khoukhi, M. Bakhouya, V. De Florio, D. El Ouadghiri, S. Latre, and C. Blondia. On the use of iot and big data technologies for real-time monitoring and data processing. *Procedia Computer Science*, 113:429–434, 2017. The 8th International Conference on Emerging Ubiquitous Systems and Pervasive Networks (EUSPN 2017) / The 7th International Conference on Current and Future Trends of Information and Communication Technologies in Healthcare (ICTH-2017) / Affiliated Workshops.

[14] Felisberto Pereira, Ricardo Correia, Pedro Pinho, Sérgio I. Lopes, and Nuno Borges Carvalho. Challenges in resource-constrained iot devices: Energy and communication as critical success factors for future iot deployment. *Sensors*, 20(22), 2020.