Addressing the five challenges in Teaching IOT

Shrutkirthi S Godkhindi
Indian Institute of Science
Bangalore, India
shrutkirthig@iisc.ac.in

T V Prabhakar
Indian Institute of Science
Bangalore, India
tvprabs@iisc.ac.in

H S Jamadagni
Indian Institute of Science
Bangalore, India
hsjam@iisc.ac.in

Abstract

In this paper, we describe five important challenges in teaching a course on IOT. The IOT course encompasses several fields of study, technologies and applications. We propose a framework to address challenges such as diverse background of students, training for solution based approach, defining core modules for the course, exposing new hardware and software tools and embracing upcoming technologies. Over the years, with the advent of new tools and platforms, several experiments were designed to address the fast changing IOT landscape. Despite obsolescence of embedded hardware, the ideas in system integration stand the test of time. Furthermore, advanced programming hitherto possible on large computers are exposed to students. One classic example is the RPC programming for simultaneous sensor data acquisition and display between ARM M7 and ARM M4. The bibliography in this paper covers most of the experiments with a detailed explanation of why a certain module had to be designed.

CCS Concepts

• **Computer systems organization → Embedded systems**.

Keywords

Low power & Energy harvesting, IOT protocols, Addressing & Localization, Internet of Things, Solution-based approach

1 Background

The origins for IOT started in the year 1999, when Kevin Ashton coined the name ‘Internet of Things’ (IOT). The foundation for this name is attributed to Radio-Frequency IDentification (RFID) technology and the first patent for batteryless passive RFID tags was granted in the year 1996. Since then its usage in several applications such as tracking, inventory, toll payments, bibs and other sports applications have seen a major success.

IOT can be applied in several applications across domains. For example, civil and mechanical engineers might be interested in its applications for structural health monitoring. An electrical engineer might be primarily interested in building a digital twin of a machine, and an electronics engineer might be interested in sensing and actuation applications. Similarly, a computer science engineer might be interested in localization algorithms. Given this diverse background of students, the ***first challenge*** for the instructor is to homogenize the learning throughout the class. Furthermore, in teaching the first course on IOT, the instructor is faced with the daunting task to focus on examples arising from the intersection of Consumer, Commercial and Industrial IOT.

The ***second challenge*** is to fast-track students who possess two to three years of industry or research lab experience. They tend to have a *Implement first* approach; while the fresh engineers have a *Learn more* approach. This exciting mix of diverse backgrounds, and varying levels of experience, ushers an enthusiastic batch of students. One common mistake committed by several students is to propose a *Technology* to solve a *Problem*, instead of following the *Problem – Tailored Solution – Right Technology* sequence. For example, the student would propose RFID if object identification and tracking is required. This is the ***third challenge*** to ensure that students follow the correct approach.

IOT can be taught either as a data-driven course or with a low-power and security based approach. Our view is that IOT should be taught with *Design* as the focus, so that it subsumes all other approaches. The ***fourth challenge*** is to enlist the core modules to tune systems for different approaches. IOT has not only seen an ever expanding scope, it has also embraced other upcoming technologies such as AI-ML, LLMs, etc. This is the ***fifth challenge***, where the instructor has to update and ensure relevance of the course and at the same time, a prerequisite set of modules have to be covered before embarking to learn and practice the upcoming areas.

This paper presents a framework for teaching a course on IOT, which addresses the above-mentioned challenges and provides a balance between theoretical foundations and hands-on practice sessions. The course is structured for postgraduate students, with an average class size of 30, and comprises 32 lecture hours and 50 hours of laboratory sessions. Assessment is carried out through a combination of theory examinations and course project implementation with equal weight. Our contributions in this work are : (a) Suggest a bridge to handle the diverse background and mixed level of experience of students. (b) Introduce a process to train students towards a solution-based approach. (c) Enlist the minimum set of core modules to teach *Design for IOT*. (d) Seamless integration with selected topics such as Digital Twin, AI-ML, LLMs, etc.

2 Challenge 1: Heterogeneous background

IOT is unique in its ability to integrate technologies across all fields and this makes it versatile and interdisciplinary course with broad relevance in today’s connected world. With such wide coverage in applications, the students crediting the course are also from diverse backgrounds. To foster such a diversified group, it is essential to structure a bridge module at the start of the course and to ensure all students have the prerequisite skills. Specifically, two *Bootcamps* of 6-hour duration each are organized to meet this normalization. Figure 1 shows the two bootcamps which are part of the bridge module.

(a) Physics & Mathematics: The goal of this module is to revise several fundamental concepts of physics and mathematics that may be applied throughout the course. As an example pyroelectric effect (PIR sensors), seebeck effect (thermo-electric generators or TEG) are discussed in detail. The basic equation of PIR is given in equation 1,

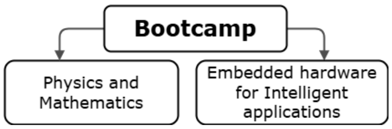

Figure 1: Bridging modules

where $\triangle Q$ represents the pyroelectric charge, P is the pyroelectric coefficient, A is surface area of pyroelectric material, and $\triangle T$ is the change in temperature. The term A should be large for high charge displacement and this is brought about by enclosing the sensor in a lens, to focus and provide large coverage area. Details related to Fresnel lens is also discussed.

$$\triangle Q = p \cdot A \cdot \triangle T \tag{1}$$

Topics in trigonometry such as Angle of Arrival (AoA), Hyperbola, Triangulation, Trilateration, Time Difference of Arrival (TDoA), Inverse trigonometric functions are briefly covered to address localization algorithms. Error estimates for these methods are also discussed. Applications of FFT is discussed in detail, e.g. data generated from IMU sensors placed on motor, pump, etc., are used for vibration analysis. A few real-world problems and embedded hardware [11], [18], [1] are provided as laboratory and take-home assignments. This includes intrusion detection, obstacle avoidance using PIR sensors [14], [24] the design of a Pedometer, with features of step-count, turn detection, and distance estimation.

(b) Embedded hardware for Intelligent applications: The goal is to familiarize students with COTS platforms, software and tools to assists in building a common base. This Bootcamp is designed to provide hands-on on several commonly used platforms such as Arduino IDE, VSCode, etc. Programming in Python and basics in C, is expected to be a prerequisite. The students are trained to work with readily available embedded hardware such as Nano 33 BLE Sense [1], Nicla Sense ME [2], Nicla Vision [3], etc., as shown in Figure 2. Further, TinyMl models are created using frameworks such as Edge Impluse [10], OpenMV [13]. Models related to gesture or object detection on the target hardware are implemented.

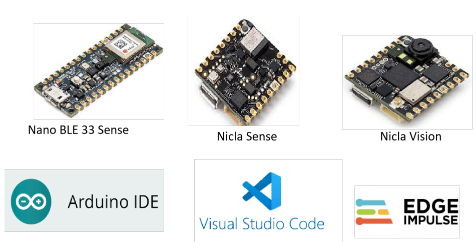

Figure 2: IOT embedded boards & software frameworks

3 Challenge 2: Diverse experience

With IOT being applied to various fields and applications, professionals and company sponsored candidates with 2-3 years of experience also pursue the course to stay aligned with the state of art. This results in a diverse group. We found through several years of teachings experience, there are mainly two approaches to handle this group.

(a) Variable scope: The idea is to challenge the experienced students to the next level for a given problem. For example, while teaching indoor localization using IMU sensors, the task for beginners can be limited to accurate step count, while the experienced students can be challenged to estimate stride length, distance and direction. Further, an opportunity to lead a discussion pertaining to an area was found to increase the satisfaction levels.

(b) Peer learning: Here, the experienced students support and encourage beginners by providing insights on a topic for enhanced learning. At the same time, beginners contribute with the latest usage methods and provide a fresh approach. For example, consider the task of implementing an autonomous navigation and obstacle avoidance system for drones. The experienced students can work on testing for different obstacle avoidance algorithms such as vision based, SLAM technique and others, whereas the beginners can focus on navigation such as checking for onboard sensors, position estimation, path planning, etc.

4 Challenge 3: Problem – Solution Approach

Firstly, students have to learn the art of mapping a given problem to the solution and its associated working condition. Furthermore, the solution may warrant quantification of parameters such as accuracy, resolution, latency, drift, range, power requirements, etc., to name a few. This exercise should be carried out *prior to proposing a technology*. In contrast, students make several assumptions and one classic example is '*location*' being synonymous to cellphone GPS. If a question such as 'Where am I?" is posed, it is easy to locate oneself today using the cellphone GPS. This question however does not mention the context and environment such as indoor, outdoor, underwater, underground or in an airplane flying at cruising altitude. If indoor or underground environment is assumed, GPS technology can only provide weak signals from a few satellites and thereby fail to localize. Further, underwater localization of a device would require data from a reference, such as a buoy.

Students should be trained to craft a "Tailored Solution" where once the problem is defined, the solution with proper working condition is proposed and later different technology alternatives are explored. Finally, after suitable evaluations, design algorithms for the appropriate candidate are implemented.

4.1 Framework towards Tailored solution

In this section, we elaborate on three broad steps required to approach the *right solution*. (1) Scope: The problem has to be well defined and the scope of be outlined with a clear boundary. (2) System design: In this step, all possible solutions are listed and evaluated with clear trade-offs for each proposed solution. (3) Implementation: the students will have to build and realize the solution, which includes the software and hardware engineering.

4.1.1 ***Scope***. Let us now take an example of a pharmacy where cartons need to be localized. Here, localization accuracy is a critical requirement (maximum of 5 - 8 cms), the area to be covered is moderately sized and there is continuous human movement. Since drugs are stacked to a height, the solution should provide the location in 3D space. The goal is to arrive at the target specification.

4.1.2 *System design*.
For the options, several techniques and localization algorithms are available such as triangulation, trilateration, multilateration, hyperbola, dead reckoning, fingerprinting and others. The students should consider parameters such as required accuracy, available data, computational constraints and environment to evaluate performance of each algorithm, to arrive at the best suited method.

In the *Design Alternatives*, selection of right technology plays the most crucial role in solution formulation. Here again, there are many technologies available for localizing, such as GPS, BLE – Beacons, NFC & RFID based methods, etc. GPS is not suitable for indoor scenario, BLE beaconing may be suitable and provides sufficient range, but requires a battery powered system for each beacon, and NFC though batteryless and reliable has a small range of a few cms. In the *evaluation* step, since EPC Gen2 RFID system supports batteryless tags, can be fixed to cartons, work indoors and has a sufficient range appears to be the candidate technology.

4.1.3 *Implementation*.
The experimental setup comprises of EPC Gen2 Dogbone passive RFID tags [21] and a UHF RFID reader [23] equipped with a dipole antenna, enabling tag detection range of several meters. The reader operates at a supply voltage of 3.3 V with a peak input current requirement of up to 1A to support power level of +27dBm. It is interfaced with a Raspberry Pi, which serves as the host for data acquisition, signal processing, and data visualization.

Figure 3 provides the code snippet of data acquisition. The code presents the steps for scanning for the tag, and to read its RSSI, frequency, timestamp, and tag identifiers. The acquired data is later used for localization algorithm. Using the pre-mapped RSSI-

```
if (responseType == RESPONSE_IS_KEEPALIVE) {
  Serial.println(F("Scanning"));
} else if (responseType == RESPONSE_IS_TAGFOUND) {

  rssi = rfidModule.getTagRSSI();
  freq = rfidModule.getTagFreq();
  timeStamp = rfidModule.getTagTimestamp();

  String tx = "", ty = "";

  for (byte x = 0; x < 2; x++) {
    char temp = char(rfidModule.msg[31 + x]);
    if (isAlphaNumeric(temp))
      tag[x] = temp;
  }
```

Figure 3: Code snippet of data acquisition from RFID module

Distance curves the distance is calculated and to further improve accuracy, the system also incorporates piecewise linear approximation. These distances serve as inputs to a triangulation algorithm, which computes the probable 2D coordinates of the tags. Figure 4 shows the implementation of 2D triangulation algorithm, where the reader moves in x-direction. Later, such multiple 2D maps at different heights are considered to create 3D localization map using InSAR (Interferometric Synthetic Aperture Radar) inspired technique.

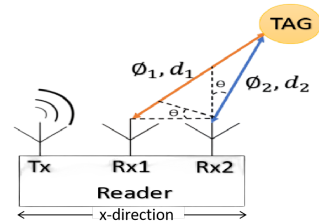

Figure 4: RFID 2D localization

5 Challenge 4: Defining Core Modules

The Rationale: The success of IOT is synonymous to the success of RFID. Passive RFID implements batteryless and reader initiated backscatter communication. The reader is the source of energy for sensing and communication. This allows RFID to support the 10+ years lifetime without maintenance. Clearly, solving the power problem is one of the "*Key*" for its success. The compact and weightless tags supported seamless blending with physical objects, leading to a wide application in different fields. Furthermore, RFID not only allows real-time data collection by sensor integration with the chip, it also provides security for data embedded in the tag; making it a secure system. To know where tags are placed, localization algorithms play an important role. Finally, reading a tag from a population of deployed tags necessitated a medium access control protocol, and one such example is the Frame Slotted Aloha applied in EPC Gen2 UHF tags.

Inspired by RFID, we derive the following insights to teach a '*First course on Internet of Things (IOT)*'. ***(a) Energy harvesting and batteryless*** technologies is an important topic. ***(b) Low power and battery efficient technologies*** are useful to build embedded IOT products that survive for over 10 years. ***(c) Sensing and processing*** is a co-design. ***(d) Data protection and privacy*** have to be addressed to ensure information is safe. ***(e) Protocols*** for medium access and communication is required. ***(f) Addressing and localization*** i.e. knowing where specific *Things* are placed is essential. Table 1 provides brief description about the learning objectives of core modules.

5.1 Energy harvesting & batteryless technologies

IOT devices deployed in the wild for continuous monitoring might be located in inhospitable terrains, making battery replacement almost impossible. Wireless charging is a possibility, perhaps a drone would be able to go close to transfer power [7], [5]. Other more realistic alternatives include energy harvesting and power management of these devices for perennial sensing [20], [12]. Although there is a significant inefficiency (less than 5%) associated with several harvesting techniques, these methods are utilized for instantaneous sensing and communication of small data e.g. rail bridge health monitoring.

The problem should be presented to students in a way that entices them to think of solutions such as opportunistic energy harvesting from the environment. For example, pressing the button on the remote can be used to convert mechanical energy into electrical energy sufficient to transmit the instruction from the remote

Table 1: Learning objectives of core modules

Core module	Learning objectives
Energy Harvesting & Batteryless Technologies	Design sustainable, batteryless IoT systems and circuits for harvesting and storage of energy. Measure harvested energy from sources such as solar, wind, vibration, temperature differentials, hydro, piezo, etc., for a given application.
Low Power & Battery Efficient Technologies	Implement hardware and software design techniques to achieve ultra-low power. Analyze power consumption based on coding practices such as blocking & non-blocking modes, low-power modes, DMA and peripherals. Chose between linear & switching regulators.
Sensing & Processing	Equip students with the skills to accurately sense, process, and interpret real-world signals in noisy environments. Build sensor circuits upon consulting datasheets. Design software and hardware solutions for drift compensation, noise suppression, bias and offset correction.
IoT Protocols	Implement IoT-specific communication protocols, considering constraints like low power, limited bandwidth, and efficiency. Demonstrate the publish-subscribe (pub-sub) and RESTful architecture. Install broker and client and analyze reliable and unreliable message delivery.
Data Protection & Privacy	Design secure and privacy-compliant IoT systems by applying key principles of data confidentiality, integrity, and authentication. Demonstrate RAM PUFs, software programs for data privacy such as differential privacy, and MUD profiles.
Addressing & Localization	Accurately localize IoT devices in large-scale, context-aware systems. Apply techniques such as hyperbola, trilateration, triangulation for localization using features like phase, AoA and TDoA. Implement unsupervised clustering techniques such as DBSCAN.

to the TV system [27]. A staircase, where the energy is generated whenever we move from one to step to another builds a musical staircase. A list of a few energy harvesting based semester-long IOT projects are shown here, (1) On AC louvers, aeroelastic flutter (wind energy) is harvested with a magnet and coil placed on either side of a vibrating belt and the energy is used to power a PIR sensor [4]. (2) Vibrations created in a moving bus can be used to generate power and drive the embedded sys tem to communicate its current location. (3) The potential energy of water stored in the overhead tank can be utilized to run a shower radio. The water pressure from the tap can be used by an impeller to generate electricity to power up low-power electronic devices [8]. Suggesting PV, Vibration, or any harvesting source by arguing the pros and cons has interested the student in examining the right solution. Basically, look at the environment and see what the available opportunities are for energy harvesting, as well as apply use it for real time applications.

5.2 Low power & battery efficient technologies

The students are assigned projects focused towards low-power design. For example, in healthcare applications, blood glucose monitoring requires to collect a sample periodically and revert to deep-sleep soon after. The system has to survive for an extended period spanning several months.

It is imperative to design this module carefully with simple and effective examples and tutorials. We need to teach a variety of important topics that are research areas in themselves. This module perhaps overlaps significantly with energy harvesting and intermittent computing. Nevertheless, there are specific aspects of this module which standout by itself.

For any device to support a 10+ years of lifetime, it becomes crucial to incorporate low-power designs for both software and hardware. The software low power design can be achieved by employing microcontrollers such as STM32L Series [25], Nordic nRF52 Series [18], etc., which provide support for sleep modes, duty cycling, fast wake-up time, and support for real time clock (RTC) [28]. Programming practices can play havoc in power consumption. Take a simple instance of a blocking statement "While loop" compared to "If Else". Basically, while blocking call statements are simpler to code, they can drain the batteries [6]. Figures 5 and 6 shows the snippet and current consumption of nRF52840 microcontroller for polling ADC using blocking and non-blocking methods. The current is measured using nRF power profiler kit II [17].

```
while (true) {
    adc_conversion_complete = false;

    // Start ADC conversion
    nrf_drv_saadc_sample();

    // BLOCKING: wait until ADC is done
    while (!adc_conversion_complete) {
        // Busy-waiting – CPU is stuck here
    }

    // Process ADC result
    printf("Blocking ADC value: %d\n", adc_result);

    nrf_delay_ms(1000);  // Wait before next sample
}
```

Figure 5: Code snippet & current consumption for Blocking ADC

Another classic example is to code for DMA to fetch data from peripherals such as UART, ADC, Digital (SPI, I2C) and other GPIO ports. To accomplish this task, the student has to be trained to consult datasheets and specifications of the microcontroller. In the datasheet one may look for important parameters such as: number of DMA channels, current consumption - uA/MHz, sleep and low power modes, operating voltage and frequency, support for DSP functions (MAC instruction), support for PWM signals for motor drivers and other parameters.

Hardware blocks in the SoC contribute significantly to the power consumption. For example, running the system using DC-DC is

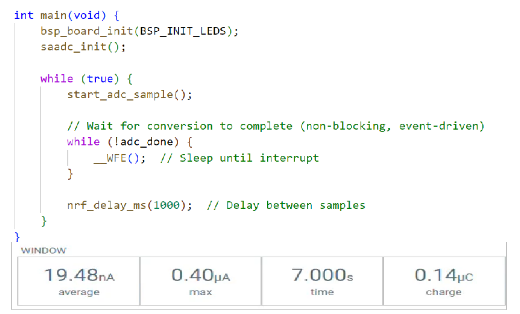

Figure 6: Code snippet & current consumption for Non-blocking ADC

energy efficient compared to LDO. RF transmission power and reception power have to be examined for improved lifetimes, all inactive GPIO ports have to be shutdown and unused peripherals have to be disabled via clock gating or power gating methods to reduce static and dynamic power consumption. ADC, power converter blocks such as DC-DC & LDO, Radio integration and other blocks that are part of the SoC have to be examined for the requirement. RAM retention can be minimized by defining essential memory block (partial RAM retention).

5.3 Sensing and processing

5.3.1 Sensing. Since a "Thing's" current state determines the action, sensing is the heart of IOT. However, data acquired from a sensor has noise by default. Thus the sensed signal has embedded noise. To handle noisy data, students are taught filter design, offset correction, drift correction and other parameters that effect the accuracy of sensed data. In ADC the offset error, gain error, DNL-INL are considered and compensated. Students are guided through various datasheets to select the appropriate sensor based on parameters such as range, sensitivity, power, resolution, operating temperature and accuracy. For example, there are a variety of IMU sensors that provide varied level of accuracy for different environments. The MPU9250 [11] MEMS low-cost IMU serves well for tilt detection and step count in low frequency indoor environment. However, aviation and drone applications demand higher temperature tolerance, high sensitivity, and accuracy, so the correct choice would be the Honeywell HG9900 [9].

5.3.2 Data Processing. The students are familiarized with different data processing techniques such as noise reduction, unit conversion, handling missing data, basics of outlier removal, smoothing filters (Moving Average, Kalman Filter), imputation, interpolation, etc. Further, data analysis and decision making techniques such as implementing threshold checks, trend detection, performing logical & statistical evaluation, conducting root cause analysis, etc., are taught. Lastly, students are exposed to different learning techniques, namely AI-ML, Gen-AI, TinyML, LLMs, etc.

5.3.3 Microcontroller. The embedded controller space is a fast changing landscape, at one end is the 8-bit controller, which can perform 'sense-and-send', and at the other end are systems that have high computational capabilities. The ARM's MAC instruction (multiply and accumulate) being a DSP function is integrated into commercial controllers. Furthermore, students are introduced to programming heterogeneous multi-core architectures. For example, in STM32 SPI, I2C, UART, DAC, ADC and on-board sensors can be assigned to either processors based on the application requirements. It facilitates parallel processing and execution, and thus enhances energy efficiency, and functional robustness as shown in [26]. One processor can manage data acquisition and processing, while the second one can simultaneously handle communication using Wi-Fi or Bluetooth, thereby improving responsiveness and reducing latency. In order to exploit the full potential of these processors inter processor communication protocols such as Remote Procedure Call (RPC) programming is taught in the class.

5.4 IOT Protocols

Communication in IOT broadly comprises two key aspects: (1) IOT specific protocols and (2) communication technologies. Students are made aware of the unique constraints in IOT systems which include limited bandwidth, low data rates, small packet sizes, and energy efficiency requirements. Protocol architectures and communication paradigms such as publish-subscribe (e.g. MQTT, AMQP) and RESTful (e.g. CoAP) are dealt in detail for their merits and associated applications [16]. Emphasis is laid on their respective performance in terms of packet overhead, packet loss & latency, and scalability.

Students are introduced to various communication technologies spanning HF, VHF and UHF ISM and licensed bands. A few examples suitable for IOT deployments are IEEE802.11ah Wi-Fi, Bluetooth Low Energy (BLE), IEEE802.15.4e-TSCH, LoRaWAN, and NB-IoT. Students explore the trade-offs in range, power consumption, throughput, and network topologies (star, mesh, peer-to-peer). Training on special tools e.g. Ubertooth [22] for debugging is imparted to the students.

5.5 Data protection and privacy

A vast amount of data is collected daily by IOT devices. This data includes personal information, activity of individuals, smart devices data, industrial data, etc. To ensure privacy, students are introduced to data confidentiality, integrity and authentication methods. Topics such as data sanitization using differential privacy, secure key management, encryption protocols, etc., are covered. Further, Manufacturer Usage Description (MUD) profiles that enhance protocol-level security by restricting device behavior on networks is also covered.

5.6 Addressing and localization

The physical location of an IOT node provides contextual information and assists in decision making. For example, the particulate matter (PM10) mass concentration for indoor and outdoor are vastly different. Also, since millions of IOT nodes are deployed across the globe, retrieving sensor data requires a unique address. Thus, location and addressing schemes such as IPv6 are covered in laboratory sessions. The sessions involve retrieving sensor data and localization algorithms to evaluate accuracy using system constraints.

6 Challenge 5: Selected topics in IoT

IOT has emerged as a building block for emerging technologies such as the Digital Twin (DT). The DT is a virtual representation of a physical object, process, or system that is continuously updated in real time with sensor data from the real world. IOT networks transmit sensor data to the DT which could be placed either at the edge or the cloud computing platforms. Physics-based DT models are by far the most accurate and heavily depend on IOT technologies for data [19]. Therefore, sensitizing students to this important topic will require drawing upon the previously discussed Challenge 4.

Another topic to discuss with students is the LLM empowerment of IOT design. Students are trained on prompt engineering concepts to extract consistent responses from LLMs. Air quality monitoring is a good example of LLM empowered design. Assume for a moment that a person carrying an IOT node equipped with a PM sensor and wishes to reach a destination location (say 5 KM away) over a route that has the least PM concentration. This would be possible if the IOT node can query LLMs that have access to world sensors and obtain information in real time to then choose the least polluting best route [15]. Another complementary approach to empower IOT is to run lightweight LLMs on Edge systems that have resource constraints. Although these may not have access to world-wide sensors, they serve important application areas such as assistive technologies, real-time language translation, robotics, healthcare, and manufacturing. Course projects in such areas seem to be make an impact on the student.

7 Course feedback

The course feedback on a few key questions collected over the past three years is as follows: *(a) Clarity of Expression and Presentation* – 4.71/5; *Pace of Teaching* – 4.67/5; *Coverage of the subjects* – 4.48/5; *Relevance of Tests and Assignments in the Class* – 4.33/5. These ratings are further affirmed by *Overall Rating* of 4.67/5.

8 Conclusions and Future work

In this paper, we have described the main challenges in teaching IOT to graduate students. A multiplier effect in learning new concepts beyond classroom theory was felt by most students, due to the continuous and rigorous process adopted for the course project. Students with prior background were fast-tracked with more challenging and deeper problems for retaining their interest and also for effective balance between fresh and experienced students.

There are several other concepts which can be covered as selected topics in IOT such as Industrial IoT (industrial–grade sensors and processors, protocols namely TiSCH, OPC-UA and other deterministic low-power protocols). Working with specialized microprocessor such as Intel Movidius, Google TPU v4, Nvidia's Jetson Nano, etc., for advanced applications should be covered. Further, Edge AI deserves to be a separate course for different applications where knowledge distillation and network pruning techniques are discussed in detail to accommodate models within the given memory space of embedded controllers. Parameter quantization such as Float – Integer conversion are demonstrated for model accuracies.

Lack of versatility of software tools is an issue rather than hardware obsolescence. While hardware evaluation platform is a quick way to start the course project, manufacturers silo the hardware with board-specific tools and this restricts end-users to work with variety of hardware platforms.

References

[1] Arduino. 2025. Nano 33 BLE Sense. Retrieved June 10, 2025 from http://docs.arduino.cc/resources/datasheets/ABX00031-datasheet.pdf
[2] Arduino. 2025. Nicla Sense ME. Retrieved June 10, 2025 from https://docs.arduino.cc/resources/datasheets/ABX00050-datasheet.pdf
[3] Arduino. 2025. Nicla Vision. Retrieved June 10, 2025 from https://docs.arduino.cc/resources/datasheets/ABX00051-datasheet.pdf
[4] CHEW:A Compact Harvesting – Energy Wind belt. 2017. Retrieved June 13, 2025 from https://labs.dese.iisc.ac.in/zenlab/past-projects/chew-a-compact-harvesting-energy-wind-belt/
[5] RF Energy Harvesting Chip. 2016. Retrieved June 13, 2025 from https://labs.dese.iisc.ac.in/zenlab/past-projects/rf-energy-harvesting-chip/
[6] embedded.com. 2019. Interfacing with modern sensors: Polled ADC drivers. Retrieved June 13, 2025 from https://www.embedded.com/interfacing-with-modern-sensors-polled-adc-drivers/
[7] Wireless Charging for Autonomous Drones. 2016. Retrieved June 10, 2025 from https://labs.dese.iisc.ac.in/zenlab/past-projects/wireless-charging-for-autonomous-drones/
[8] Micro-Hydro Generator. 2017. Retrieved June 13, 2025 from https://labs.dese.iisc.ac.in/zenlab/past-projects/micro-hydro-generator/
[9] Honeywell 2018. *HG9900 Inertial Measurement Unit*. Honeywell. https://aerospace.honeywell.com/content/dam/aerobt/en/documents/learn/products/sensors/brochures/N61-1638-000-000-hg9900inertialmeasurementunit-bro.pdf.
[10] Edge Impulse. 2025. Software. Retrieved June 10, 2025 from https://edgeimpulse.com/
[11] InvenSense 2015. *MPU-9250 Product Specification*. InvenSense. https://invensense.tdk.com/wp-content/uploads/2015/02/PS-MPU-9250A-01-v1.1.pdf.
[12] Aman Kansal and et al. 2007. Power management in energy harvesting sensor networks. *ACM Transactions on Embedded Computing Systems (TECS)* 6, 4 (2007), 32–es.
[13] OpenMV. 2025. Software. Retrieved June 10, 2025 from https://openmv.io/
[14] Panasonic. 2023. Grid-EYE Datasheet. https://industrial.panasonic.com/cdbs/www-data/pdf/ADI8000/ast-ind-241606.pdf [Accessed: November 16, 2023].
[15] Least PM path. 2025. Retrieved June 19, 2025 from https://github.com/Shrutkirthi/AirQualityMonitoring
[16] IOT Protocol. 2025. Retrieved June 25, 2025 from https://github.com/DeekshaPRao/IOT_protocol_start
[17] Nordic Semiconductor. 2022. Power Profiler Kit II. Retrieved June 18, 2025 from https://www.nordicsemi.com/Products/Development-hardware/Power-Profiler-Kit-2
[18] Nordic Semiconductors. 2025. nRF52840. Retrieved June 10, 2025 from https://docs.nordicsemi.com/bundle/ps_nrf52840/page/keyfeatures_html5.html
[19] Karan Shah, TV Prabhakar, CR Sarweshkumar, SV Abhishek, et al. 2021. Construction of a digital twin framework using free and open-source software programs. *IEEE Internet Computing* 26, 5 (2021), 50–59.
[20] Gaurav Singh and et al. 2013. A tuned rectifier for RF energy harvesting from ambient radiations. *AEU-International Journal of Electronics and Communications* 67, 7 (2013), 564–569.
[21] Smartrac. 2021. Dog bone RFID tag datasheet. Retrieved June 10, 2025 from https://rfid.averydennison.com/content/dam/rfid/en/products/rfid-products/data-sheets/datasheet-Dogbone-UCODE-8.pdf
[22] Sparkfun 2015. *Ubertooth One specification*. Sparkfun. https://mm.digikey.com/Volume0/opasdata/d220001/medias/docus/2508/WRL-10573_Web.pdf.
[23] Sparkfun. 2018. *M6E Nano*. Retrieved June 10, 2025 from https://cdn.sparkfun.com/assets/4/e/5/5/0/SEN-14066_datasheet.pdf
[24] Ultrasonic Ranging Module HC SR04. 2018. Retrieved June 20, 2025 from https://cdn.sparkfun.com/datasheets/Sensors/Proximity/HCSR04.pdf
[25] STMicroelectronics. 2024. *STM32L476xx*. Retrieved June 15, 2025 from https://www.st.com/resource/en/datasheet/stm32l476je.pdf
[26] Felix Sutton and et al. 2015. Bolt: A stateful processor interconnect. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*. 267–280.
[27] Wireless Switch. 2016. Retrieved June 10, 2025 from https://labs.dese.iisc.ac.in/zenlab/past-projects/wireless-switch/
[28] Wake up Radio. 2019. Retrieved June 13, 2025 from https://labs.dese.iisc.ac.in/zenlab/past-projects/wake-up-radio/