Chapter I: Overview of Embedded Systems Projects Conducted at Our School

I Introduction

I.1 Project 1: Drone for Air Quality Monitoring

The following project, carried out as part of the Embedded Systems course, focuses on the design and implementation of a drone dedicated to air quality monitoring. The idea emerged from the need to measure, in a flexible and real-time manner, the concentration of pollutant gases in different areas, particularly in locations that are difficult to access with fixed monitoring stations. Given the high cost and limited mobility of conventional measurement systems, the proposed solution was to design an autonomous drone capable of collecting environmental data through embedded sensors.

The main objective of this project is to develop a complete embedded system for detecting and transmitting parameters such as CO_2, CO, and methane (CH_4) concentrations, as well as ambient temperature and humidity. The drone aims to provide a dynamic mapping of air quality, thereby contributing to environmental monitoring efforts and pollution risk prevention.

The project provided an opportunity to explore several key embedded systems concepts, including:

- **Motor control:** ensuring drone stability and trajectory management through precise control of brushless motors.
- **Data acquisition and processing:** real-time reading and processing of values from gas and environmental sensors.
- **Wireless communication:** data transmission to a ground station using the ESP32's built-in Wi-Fi or Bluetooth capabilities.
- **Autonomous navigation:** integration of a GPS and altimeter for trajectory tracking and geolocation of measurements.

At the core of the system is an **ESP32** microcontroller, which manages both the sensor data and flight control while providing wireless connectivity. MQ-series sensors (MQ-2, MQ-7, MQ-135) are used to detect gas concentrations, while a DHT22 sensor measures temperature and humidity for data correction and environmental context.

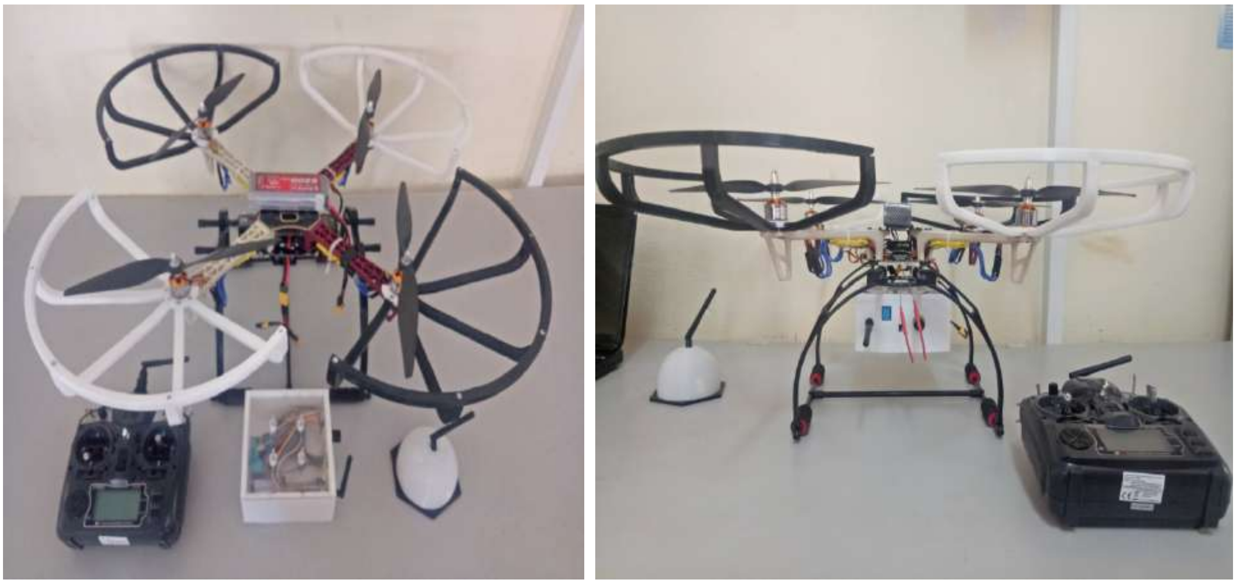


Figure I. 1 - Project 1: Drone for Air Quality Monitoring

On the software side, the ESP32 runs a program that continuously reads sensor values, transmits the collected data, and triggers alerts when threshold levels are exceeded. The user interface, developed on a PC or mobile application, allows real-time visualization of the measured parameters and displays the drone's flight path on a map.

The prototype includes the mechanical structure of the quadcopter, the electronic control board, the sensor module, and the communication system. Initial tests confirmed the drone's flight stability and the reliability of wireless data transmission. The results validated the technical feasibility of the system and its potential for real-world environmental monitoring applications.

This project exemplifies the integration of **mechanical**, **electronic**, **data processing**, and **automation** components within a complex embedded system. It also highlights the interdisciplinary nature of embedded engineering, combining control theory, intelligent sensing, and wireless communication.

Future improvements could focus on:

- optimizing the flight path for more uniform coverage of monitored zones,
- integrating artificial intelligence techniques for automatic pollution source detection,
- and deploying a cooperative drone network for large-scale environmental monitoring.

This work contributes to the development of innovative technological solutions for environmental protection and intelligent air quality management, fully aligned with the principles of sustainable development.

I.2 **Project 2:** Design and Implementation of an Embedded System for Fire Safety

This project, developed as part of the Embedded Systems course, focuses on the design and implementation of an intelligent fire detection and monitoring system. The idea originated from the growing need for autonomous, reliable, and connected devices to prevent and quickly manage fire risks in industrial and domestic environments.

The system is based on a distributed embedded architecture using ESP32 boards connected via a Wi-Fi network and communicating through the MQTT (Message Queuing Telemetry Transport) protocol. This lightweight and efficient protocol enables real-time data exchange between the various detection nodes and the monitoring interface developed using Node-RED.

The main objective of the project is to provide early fire detection by simultaneously measuring several physical parameters:

- **Flame presence**, detected by an infrared flame sensor;
- **Gas concentration** (smoke or combustible gases), measured by an **MQ-2** sensor;
- **Ambient temperature and humidity**, monitored by a **DHT11** sensor.

Each monitored zone is equipped with an **ESP32** module that collects sensor data and publishes it to an **MQTT broker** (Mosquitto) installed on a PC. This broker centralizes the information and redistributes it to the Node-RED dashboard, where the user can visualize the real-time status of each zone and receive automatic alerts in case of smoke, flame, or abnormal temperature detection.

The system is organized in a modular structure, allowing the simultaneous monitoring of several independent zones (Zone 1, Zone 2, Zone 3). Each module acts as an MQTT client (ClientMQTT-1, ClientMQTT-2, etc.) and communicates with the server using specific **topics**, such as:

- device1/temp for temperature,
- device1/hum for humidity,
- device1/gas for gas concentration,
- device1/flame for flame detection,
- and device1/relay for remote control of an alarm or fire suppression system.

From a hardware perspective, the system includes:

- an **ESP32** board responsible for data acquisition, processing, and transmission;
- a set of sensors (flame, gas, temperature, and humidity);
- a relay for activating a visual or audible alarm;
- and a **Node-RED** supervision interface connected to the **Mosquitto** broker.

The figures below illustrate the **functional diagram** of the system and the **hardware prototype** composed of the three detection zones. Each zone is mounted on a breadboard and connected to the Wi-Fi network to transmit real-time measurements.

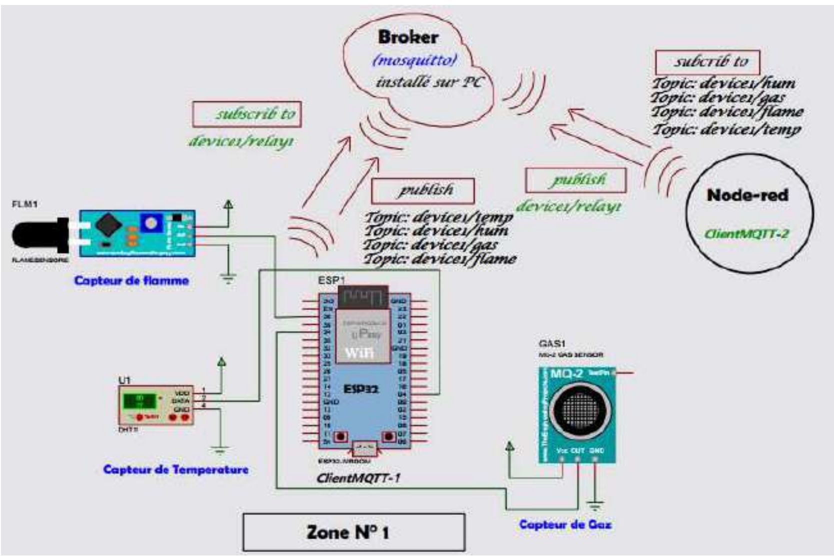
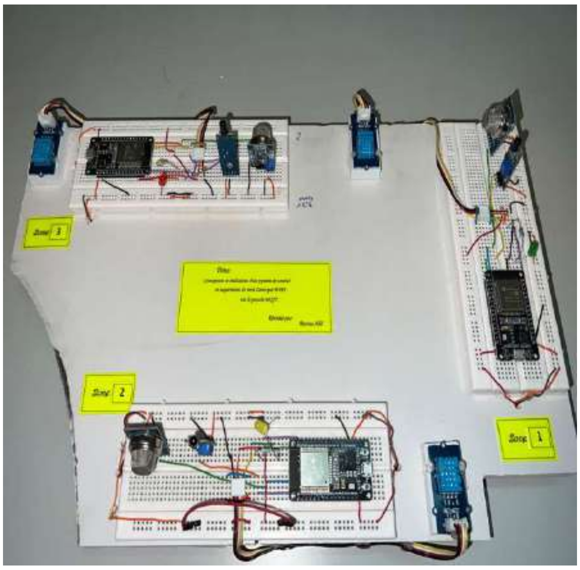
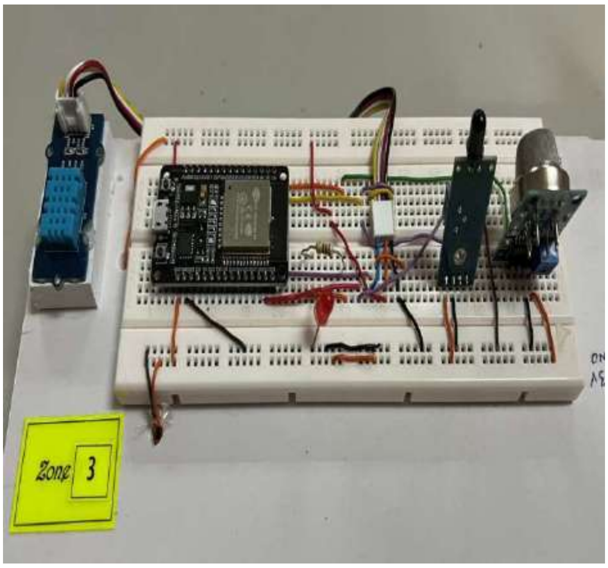


Figure I. 2 - Project 2: Design and Implementation of an Embedded System for Fire Safety

Initial testing demonstrated the system's ability to quickly detect thermal or gaseous anomalies and instantly transmit alerts to the monitoring interface. The results confirmed the reliability of the MQTT protocol, the responsiveness of the ESP32 microcontroller, and the stability of the communication network.





This project concretely illustrates the integration of embedded electronics, the Internet of Things (IoT), and intelligent communication systems for infrastructure protection. It highlights the advantages of a modular and connected approach to safety systems, promoting centralized monitoring and rapid response in the event of an incident.

I.3 Project 3: Automated Attendance Management System

This project demonstrates a practical application of RFID (Radio Frequency Identification) technology in the educational domain. It involves the design and implementation of an automated attendance management system aimed at accurately monitoring student attendance within academic institutions. The main objective is to facilitate the identification of absentee students and provide reliable indicators of attendance for teachers and administrators.

The system architecture is composed of three integrated components:

1. **Electronic Component:** Each student carries an RFID card. Upon entering the classroom, the card is detected by an RFID reader connected to an Arduino UNO microcontroller. The reader captures the radio frequency signal emitted by the card and sends the student's ID to the Arduino for preliminary processing, such as activation and attendance recording.
2. **Telecommunication Component**: The attendance data collected by the Arduino are transmitted to a central server via an Ethernet module. This network link acts as the communication bridge between the physical hardware and the information system.
3. **Software Component**: The software part includes a server, a database, and a web application. The database stores student, teacher, course, and schedule data. The web interface—developed using PHP/MySQL and HTML/CSS—enables real-time attendance tracking, report generation, and statistical analysis. It provides multi-platform access from computers, tablets, and smartphones.

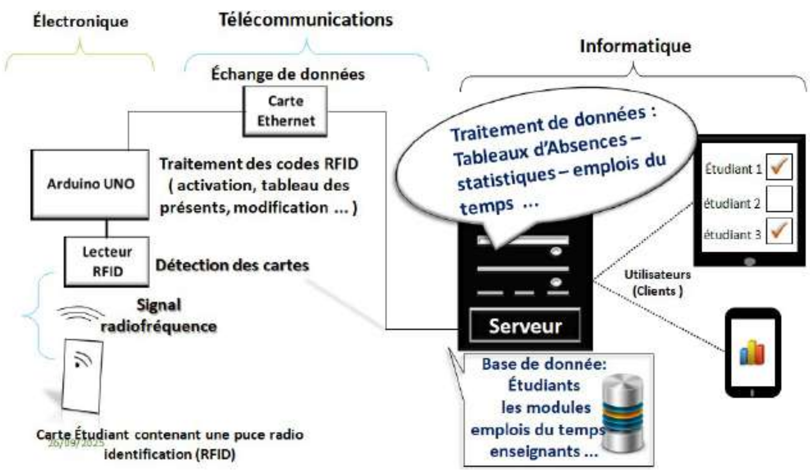


Figure I. 3 - General architecture of the automated attendance management system based on RFID technology.

This figure illustrates how the RFID card is detected by the reader connected to the Arduino, the data are then transmitted through the Ethernet module to the server, which processes and stores the information in the database. The web application allows teachers and administrators to visualize attendance in real time and generate statistical reports. The diagram highlights the synergy between electronics, telecommunications, and computer science in developing a reliable embedded system.

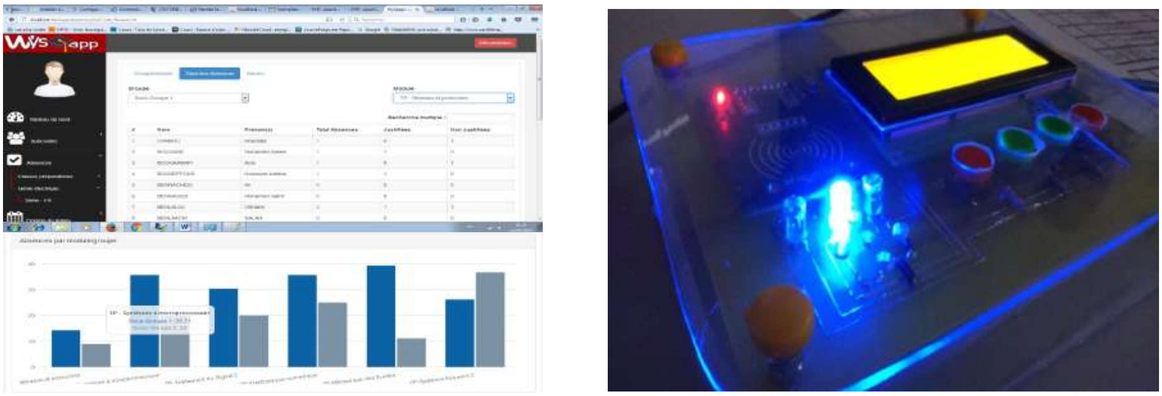


Figure I. 4 - Developed hardware prototype and associated software interface for attendance management and monitoring.

The prototype demonstrates the physical implementation using Arduino and the RFID reader, while the web interface showcases the available functionalities for attendance tracking, schedule visualization, and statistical reporting. Together, they validate the feasibility and practical relevance of the proposed system in real-world educational settings.

This work was presented under the title: "Design and Implementation of an Embedded System for Effective Attendance Management", at the NewMat'21 – 1st International Conference: New Trends on Innovative Construction Materials, held at ESSA Tlemcen (Algeria), on March 22–23, 2022.

I.4 Project 5: Development of a 3D Printer – Version 1

The second project conducted as part of the embedded systems course focused on the design and implementation of a first-version 3D printer. The idea emerged from the need to produce custom plastic components for scientific projects while significantly reducing the high costs associated with commercial 3D printing services. As engineering students, the proposed solution was to build the printer from scratch using readily available components and open-source software.

The main objective was to develop a fully functional 3D printer capable of manufacturing plastic prototypes using materials such as PLA and ABS. The project provided an opportunity to explore multiple embedded system concepts, particularly:

- **Motor control**: precise positioning through stepper motors for the X, Y, and Z axes.
- **Thermal regulation:** control of the heating extruder and heated bed.
- **System interfacing**: communication between the microcontroller and the user interface for process monitoring and control.

At the core of the system lies an Arduino Mega microcontroller equipped with a RAMPS 1.4 shield and stepper motor drivers to manage movement. Limit switches were integrated to calibrate the axes and ensure accurate positioning.

On the software side, the Marlin firmware was installed on the Arduino to interpret G-code commands generated by slicing software such as Cura or Slic3r. This allows the user to go seamlessly from a 3D model designed on a computer to a physical printed object, with real-time monitoring and control of printing parameters through a PC interface.

The prototype, shown in Figure 7, includes the mechanical structure, the electronic control board, and the power modules. This initial version successfully achieved the printing of small test objects, confirming the technical feasibility of the system. The project also demonstrated the pedagogical value of 3D printing, as it enables rapid prototyping—turning ideas into tangible models—while reducing manpower and tooling requirements.

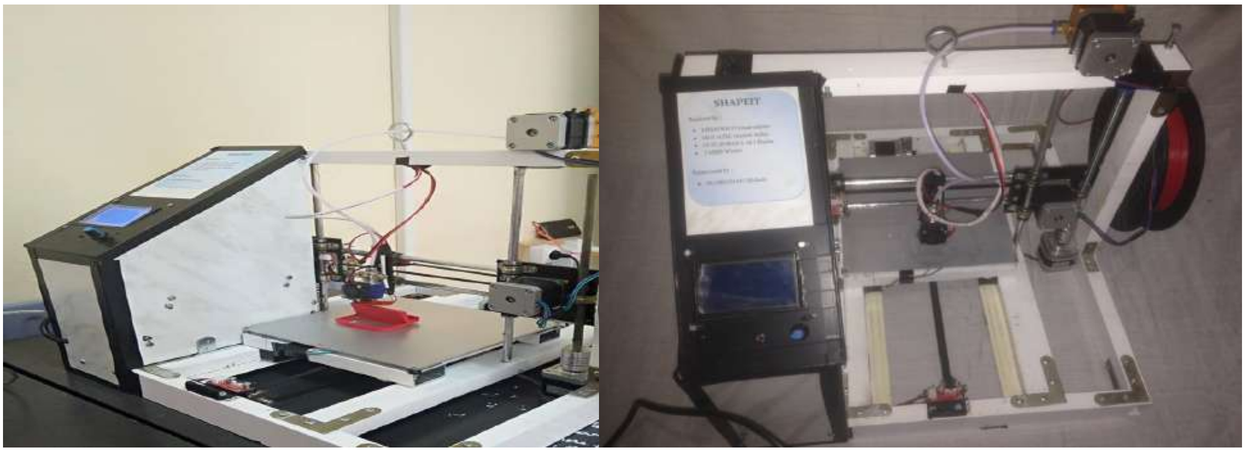


Figure I. 5- Prototype of the 3D printer (Version 1) developed by engineering students, showing the mechanical frame, control electronics, and software interface.

This first version represents a foundational step in the development of a 3D printer. Future improvements could focus on increasing print speed, enhancing print quality, and improving overall reliability. Beyond its technical aspects, this project exemplifies the integration of mechanical, electronic, and software components within a complex embedded system. It also highlights the interdisciplinary nature of embedded engineering, combining control theory, automation, and digital manufacturing.

This work was presented in the following publication: H. Megnafi, O. Ayad, W. Tabib, A. A. Mouaziz, R. Ould Babaali, I. Medjhoud, "Improved Printing Time by Changing the Mechanical Part of the 3D Printer, Embedded System Application," NewMat'21 – 1st International Conference: New Trends on Innovative Construction Materials, ESSA-Tlemcen (Algeria), March 22–23, 2022.

I.5 Project 6: Development of a 3D Printer – Version 2

The second version of the 3D printer was designed with the goal of improving the performance achieved with the initial prototype. This enhanced version integrates a more efficient mechanical architecture, resulting in a threefold increase in printing speed and greater precision in the fabrication of printed parts.

The control system is built around an Arduino Mega 2560 microcontroller combined with the RAMPS 1.4 shield, which serves as the interface between the controller and the actuators. The printer's movement along the X, Y, and Z axes is handled by NEMA 17 stepper motors, driven by A4988 drivers that regulate the current and ensure precise motion control.

On the mechanical side, the structure was redesigned to minimize vibration and improve rigidity. A heated bed was added to enhance the adhesion of printed parts during the initial layers, while the hotend nozzles reach temperatures suitable for a variety of materials such as PLA and ABS. Limit switches were also integrated to establish reference positions for each axis, allowing for automatic calibration.

From an interface and autonomy perspective, a LCD screen with an SD card reader was implemented, enabling standalone operation without the need for a computer connection during printing. A 12V – 20A power supply provides sufficient energy for the heated bed, motors, and extruder heaters. The system operates under the Marlin firmware, configured to support dual-extruder management and optimized axis movement. The G-code files, generated through slicing software such as Cura or Repetier-Host, are directly interpreted by the Arduino, which controls the full printing sequence autonomously.

These hardware and software upgrades significantly reduced printing time while enhancing surface quality, consistency, and precision. This project demonstrates how refining both mechanical and electronic architectures can dramatically improve the performance of a complex embedded system such as a 3D printer.

Figure I. 6 - 3D Printer – Version 2.

This work was presented in the following publication: H. Megnafi, O. Ayad, W. Tabib, A. A. Mouaziz, R. Ould Babaali, I. Medjhoud, "Improved Printing Time by Changing the Mechanical Part of the 3D Printer, Embedded System Application," NewMat'21 – 1st International Conference: New Trends on Innovative Construction Materials, ESSA-Tlemcen (Algeria), March 22–23, 2022.

I.6 Project 7: Design and Implementation of an Autonomous Watering System Based on PIC18F452

This project focuses on the design and realization of an autonomous irrigation system aimed at optimizing water management for agricultural and gardening applications. The system is entirely self-powered, operating on solar energy through a solar panel coupled with a rechargeable battery, thus ensuring complete energy autonomy. The PIC18F452 microcontroller serves as the central processing unit, managing data acquisition, decision-making, and the control of various sensors and actuators.

The system integrates soil moisture sensors and a LM35 temperature sensor to continuously monitor environmental and soil conditions. Based on predefined thresholds, the microcontroller automatically controls electrovalves to activate watering when necessary. This automated process ensures that plants receive the right amount of water while preventing waste, contributing to more efficient and sustainable irrigation management.

The user interface includes a 16×2 LCD display and push buttons, allowing users to select operation modes (automatic or manual) and configure system parameters. Furthermore, a Wi-Fi communication module enables remote supervision and control, allowing the user to monitor environmental data and system status via a smartphone or computer.

Figure I. 7 - Block diagram of the autonomous watering system.

The block diagram illustrates the main components of the system: the solar energy unit (panel and battery) supplies power to the control circuit; the sensors provide environmental data to the PIC18F452, which processes the inputs and drives the electrovalves accordingly. The LCD display and Wi-Fi module facilitate local and remote interaction, respectively.

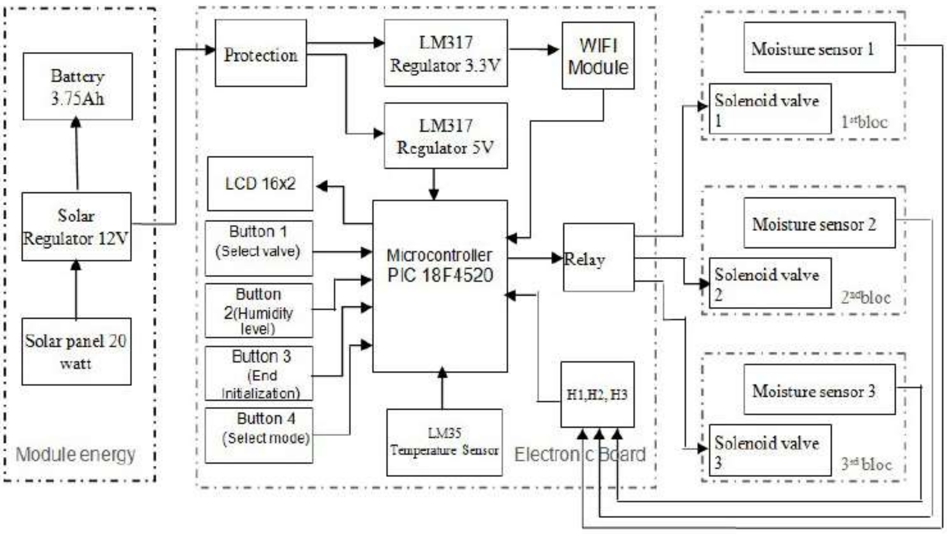


Figure I. 8 - Prototype of the autonomous watering system.

The developed prototype demonstrates the integration of hardware and software subsystems in a compact, functional design. It validates the feasibility of an energy-autonomous, intelligent irrigation controller capable of real-time adaptation to environmental changes.

This project exemplifies how embedded systems can be leveraged to address sustainability challenges by combining renewable energy, automation, and intelligent control.

This work was presented in the following publications:

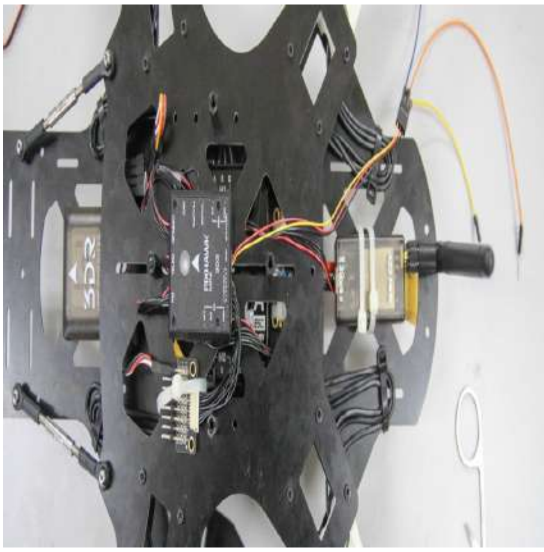
- A. Chellal, H. Megnafi, A. Benhanifia, Design and Conception of an Autonomous Watering System, Garden (a Multi-Application Watering System), NewMat'21 – 1st International Conference: New Trends on Innovative Construction Materials, ESSA-Tlemcen (Algeria), March 22–23, 2022.
- H. Megnafi, A. A. Chellal, A. Benhanifia, *Flexible and Automated Watering System Using Solar Energy*, in *Artificial Intelligence and Renewables Towards an Energy Transition 4*, Springer International Publishing, 2021, pp. 747–755.

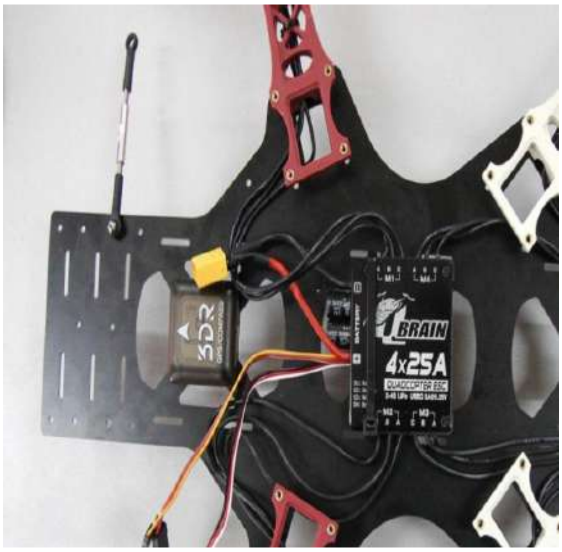
I.7 Project 8: Configuration and Assembly of a Quadrotor Drone for Telecommunications Applications

This project was developed as part of the embedded systems course and focuses on the design, configuration, and assembly of a quadrotor drone intended for applications in the field of telecommunications. The main objective is to develop a drone capable of carrying out surveillance, mapping, and radio site inspection missions in areas that are difficult to access, while leveraging modern embedded technologies and wireless communication protocols.

The developed system is based on a modular embedded architecture, integrating the following components:

- Pixhawk 3DR flight control board: responsible for drone stabilization, management of inertial sensors (gyroscope, accelerometer, barometer), and execution of control and navigation algorithms.
- Electronic Speed Controllers (ESCs) for regulating the four brushless motors.
- GPS module for positioning and autonomous navigation.
- Onboard sensors (altitude, temperature, voltage, current) for in-flight monitoring.
- Radio telemetry system for bidirectional communication between the drone and the ground station.
- Camera or IoT module for real-time transmission of images or monitoring data.





The mechanical assembly is built on a carbon-fiber X4 frame (quadrotor type), ensuring an optimal balance between stability, lightness, and robustness. The drone is powered by a Li-Po battery and controlled either via a 2.4 GHz remote controller or through a ground control station using Mission Planner or QGroundControl software.

The software configuration phase includes:

- calibration of sensors (gyroscope, accelerometer, compass);
- tuning of PID parameters for flight stabilization;
- programming of autonomous missions (waypoints, altitude, and speed);
- integration of a telemetry module for real-time flight data supervision.

Experimental tests validated the flight stability, communication reliability, and the drone's ability to perform autonomous missions. These results demonstrate the system's potential for use in telecommunication site monitoring, radio signal mapping, and preventive maintenance in complex or remote environments.

I.1.1 Project 9: Development of an Intelligent Urban Lighting System

This project focuses on the design and implementation of an IoT-based smart public lighting system aimed at significantly reducing energy consumption while improving operational efficiency and sustainability in urban environments. The developed prototype utilizes an ESP8266 NodeMCU microcontroller, which manages and coordinates the operation of the various modules within the system.

The system’s power supply is provided by a photovoltaic solar panel, connected to a voltage regulator and a charging circuit that stores energy in a lithium battery, ensuring complete energy autonomy. This solar-based design not only supports renewable energy use but also enables continuous operation even in areas without access to the electrical grid.

A PIR (Passive Infrared) motion sensor is integrated for presence detection, allowing the system to automatically adjust lighting intensity based on pedestrian or vehicle movement. When no movement is detected, the light operates at a reduced intensity to conserve energy; it brightens immediately upon motion detection. A 16×2 LCD display provides real-time system status and data visualization, while a high-efficiency LED module is used for illumination.

The ESP8266 microcontroller, equipped with built-in Wi-Fi capability, ensures IoT connectivity and remote supervision. This allows monitoring and control through an online dashboard, enabling urban management authorities to track energy consumption, system performance, and maintenance needs in real time.

This smart lighting architecture integrates renewable energy, intelligent control, and wireless communication, resulting in a system that is energy-efficient, autonomous, and well-suited for smart city applications.

Figure I. 9 - Simplified schematic of the intelligent urban lighting system and the developed prototype.

The figure illustrates the system's functional structure, showing the interaction between the solar power subsystem, the control and sensing units (ESP8266 and PIR sensor), and the LED lighting module. It also depicts how the IoT layer enables remote communication and system monitoring.

This work was presented in the following publication: Imen Souhila Bousmaha, Hicham Megnafi, Hamza Benhadouga, Imene Hemarid, IoT Applications in Smart Public Lighting Management, The International Conference on Applied Science and Engineering (ICASE-22).

I.8 Project 10: Design and Implementation of a Control System for a Hybrid Electrical Cabinet through a Remote IoT-Based Interface

This project, carried out as part of the electrical engineering program, focuses on the design and implementation of a remote supervision and control system for a hybrid electrical cabinet. The system combines renewable energy sources (photovoltaic and wind) with an auxiliary source (diesel generator) to ensure a reliable, continuous, and optimized power supply for isolated sites such as telecommunication stations or autonomous industrial facilities.

The main concept of this project lies in the integration of Internet of Things (IoT) technologies within a hybrid energy infrastructure. This approach enables real-time monitoring of system performance, remote control of power equipment, and energy management optimization through an intelligent, connected interface.

The overall architecture of the system consists of three main electrical cabinets:

- **Cabinet 01**: dedicated to the photovoltaic (PV) source,
- **Cabinet 02**: connected to the wind (EO) source,
- **Cabinet 03**: responsible for load distribution, protection, and control, while integrating the IoT supervision and control module.

Each cabinet is equipped with ESP32 boards that acquire, process, and transmit both electrical and environmental data to a remote server via Wi-Fi. The HTTP/MQTT protocol is used for data exchange between the microcontrollers and the ThingSpeak platform, which acts as a centralized cloud server.

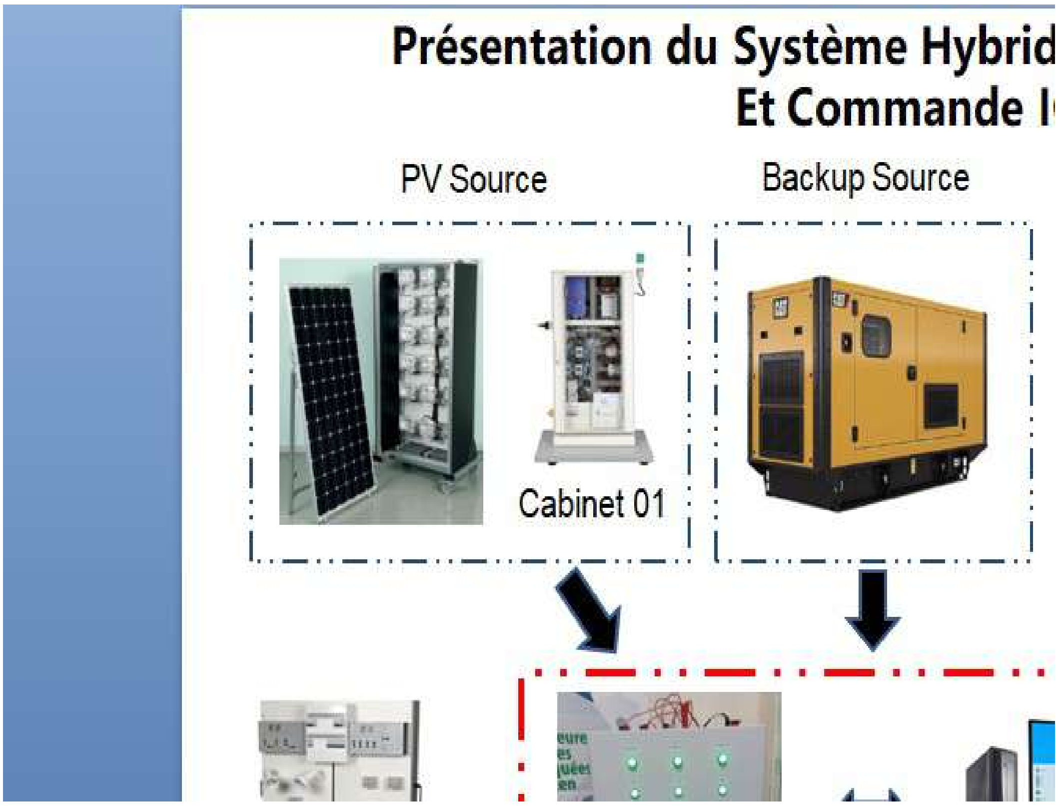
The system includes several functional blocks:

- a State of Charge (SoC) supervision block for monitoring the batteries and charge regulators;

- a current supervision block for measuring the currents produced by each source and consumed by the loads using INA219 sensors;
- an environmental monitoring block that records external parameters such as temperature, wind speed, and solar irradiation through sensors connected to an ESP32 board;
- and finally, a relay and contactor control block, allowing both automatic and manual switching between power sources.

Remote supervision and control are managed through two main software interfaces:

- Node-RED, which provides real-time graphical supervision and manual remote control of the relays via virtual switches, while integrating weather data from the Open-Meteo platform;
- MATLAB GUI, which offers an interactive interface for data visualization and automated decision-making based on predefined scenarios that consider total current demand, state of charge (SoC), and weather conditions.



*

The experimental results confirmed the reliability of the system in the simultaneous supervision of the three sources and loads, the stability of IoT communications, and the responsiveness of manual and automated control commands.