

DESIGN AND EXPERIMENTAL VALIDATION OF AN IoT-RPA-BASED SMART REFRIGERATOR INVENTORY SYSTEM

Lyudmila Samchuk¹, Yuliia Povstiana², Nataliia Lishchyna², Mykyta Ponomarenko³

¹Lutsk National Technical University, Department of Applied Mechanics and Mechatronics Lutsk, Ukraine, ²Lutsk National Technical University, Department of Software Engineering, Lutsk, Ukraine, ³Lutsk National Technical University, Department of Software Engineering, Lutsk, Ukraine

Abstract: This paper presents the design and experimental validation of an integrated IoT-RPA cyber-physical system for automated food inventory monitoring and replenishment in a smart refrigerator. The proposed solution combines weight-based IoT sensing with robotic process automation to form a closed-loop "measurement-decision-action" architecture that minimizes user involvement. The hardware subsystem integrates a strain-gauge load cell, an HX711 analogue-to-digital converter, and a WeMos D1 (ESP8266) microcontroller to measure product weight and transmit telemetry via Wi-Fi to a local Flask-based server. Measurement data are stored in CSV format and processed by a Robotic Process Automation (RPA) module implemented in Microsoft Power Automate Desktop, which initiates a digital ordering scenario when the product weight falls below a predefined threshold. Experimental validation under conditions close to real household operation demonstrated a mean absolute error below 2% and a coefficient of variation under 5%, as well as reliable transmission of more than 10,000 telemetry records without data loss. The RPA subsystem was evaluated through 120 execution cycles over seven days, achieving a success rate of 93.3% with an average execution time of 27.4 s under real web-interface constraints. The results confirm the feasibility, stability, and scalability of the proposed IoT-RPA architecture for autonomous household inventory management.

Keywords: IoT, smart refrigerator, load cell, ESP8266, Flask, robotic process automation

PROJEKT I WALIDACJA EKSPERYMENTALNA INTELIGENTNEGO SYSTEMU ZARZĄDZANIA ZAPASAMI W LODÓWCE OPARTEGO NA TECHNOLOGII IoT-RPA

Streszczenie: W niniejszym artykule przedstawiono projekt oraz eksperymentalną walidację zintegrowanego systemu cyfrowo-fizycznego opartego na technologii IoT i RPA, służącego do automatycznego monitorowania stanów magazynowych żywności i uzupełniania zapasów w inteligentnej lodówce. Proponowane rozwiązanie łączy czujniki IoT mierzące masę z automatyzacją procesów robotycznych, tworząc architekturę typu "pomiar-decyzja-działanie" w pętli zamkniętej, która minimalizuje zaangażowanie użytkownika. Podsystem sprzętowy obejmuje czujnik wagowy z tensometrem, przetwornik analogowo-cyfrowy HX711 oraz mikrokontroler WeMos D1 (ESP8266), służące do pomiaru masy produktów i przesyłania danych telemetrycznych przez Wi-Fi do lokalnego serwera opartego na Flasku. Dane pomiarowe są przechowywane w formacie CSV i przetwarzane przez moduł automatyzacji procesów robotycznych (RPA) zaimplementowany w aplikacji Microsoft Power Automate Desktop, który uruchamia scenariusz zamówienia cyfrowego, gdy waga produktu spadnie poniżej z góry określonego progu. Walidacja eksperymentalna w warunkach zbliżonych do rzeczywistej eksploatacji w gospodarstwie domowym wykazała średni błąd bezwzględny poniżej 2% i współczynnik zmienności poniżej 5%, a także niezawodną transmisję ponad 10 000 rekordów telemetrycznych bez utraty danych. Podsystem RPA został poddany ocenie w ramach 120 cykli wykonawczych w ciągu siedmiu dni, osiągając wskaźnik powodzenia na poziomie 93,3% przy średnim czasie wykonania wynoszącym 27,4 s w rzeczywistych warunkach ograniczeń interfejsu internetowego. Wyniki potwierdzają wykonalność, stabilność i skalowalność proponowanej architektury IoT-RPA do autonomicznego zarządzania zapasami w gospodarstwie domowym.

Słowa kluczowe: IoT, inteligentna lodówka, czujnik masy, ESP8266, Flask, automatyzacja procesów robotycznych

Introduction

The rapid advancement of digital technologies, particularly the Internet of Things (IoT), has significantly transformed approaches to automation not only in industrial environments but also in everyday household systems. The key challenge is not merely the development of technically advanced devices, but their effective integration into daily processes in a manner that reduces user involvement and operational overhead.

One of the promising application areas of IoT in household environments is automated food inventory management. Conventional approaches to managing household food supplies rely heavily on manual monitoring, periodic inspections, and user-driven decision-making, which are prone to human error and inefficiencies. Existing smart refrigerator solutions typically employ either manual data input or vision-based recognition techniques. However, vision-based systems are sensitive to lighting conditions, product placement, and packaging variability, which can negatively affect reliability and measurement accuracy. These limitations indicate the need for alternative sensing approaches that provide objective and repeatable measurements.

Weight-based sensing represents a robust and hardware-efficient method for monitoring product availability in closed environments such as refrigerators. Load-cell sensors offer direct measurement of mass changes and are largely independent of visual disturbances. Despite their advantages, most existing implementations of weight-based inventory systems are limited to data acquisition and notification functionalities, without providing a fully autonomous mechanism for initiating replenishment actions. In parallel, Robotic Process Automation (RPA) has gained increasing attention as a technology for automating rule-based digital processes, particularly in procurement, order handling, and supply chain management. RPA enables software

agents to interact with existing web interfaces and information systems without requiring dedicated application programming interfaces. However, current studies on RPA predominantly address enterprise-level applications and rarely consider its integration with physical sensing systems in domestic environments.

The scientific novelty of this study consists in the following contributions: a closed-loop cyber-physical architecture is proposed that directly links continuous weight-based IoT sensing with autonomous digital actions implemented via robotic process automation (RPA). The main contributions of this work can be summarized as follows: Development of a closed-loop IoT-RPA cyber-physical architecture for household inventory management. Experimental validation of a weight-based sensing subsystem operating under refrigerator conditions. Quantitative evaluation of RPA performance using execution success rate, latency, and MTBF metrics. Demonstration of autonomous interaction between physical sensing and digital ordering workflows.

From a research perspective, this work primarily represents an engineering-oriented contribution focused on the design, integration, and experimental validation of a cyber-physical IoT-RPA system, rather than the development of a fundamentally new theoretical method. However, it provides a structured architectural approach and quantitative evaluation framework that can support further scientific research in autonomous household inventory systems.

1. Literature review

Research on intelligent inventory management can be categorized into four main directions relevant to this study: IoT-based inventory monitoring systems, weight-based sensing approaches, robotic process automation (RPA) in purchasing and order handling, and smart kitchen and online grocery ecosystems.



Although each of these areas has been extensively studied, their integration into a unified closed-loop household cyber-physical system remains limited.

The first group includes IoT-based inventory monitoring systems that rely on low-cost microcontrollers, networked sensors, and web or mobile interfaces. Previous studies demonstrate the feasibility of real-time stock monitoring and notification in domestic environments [6, 8]. These systems effectively collect and visualize data; however, their functionality is typically limited to user alerts. As a result, decision-making and replenishment actions still depend on manual user intervention.

The second research direction focuses on more advanced monitoring environments, particularly in retail applications. For example, smart shelf systems combine load cells with computer vision to track product availability and customer behaviour [5]. Such hybrid solutions offer higher detection capability and contextual awareness. At the same time, they require complex sensing infrastructure, camera deployment, and increased computational resources. These requirements limit their applicability in compact household environments, where cost, simplicity, and ease of installation are critical factors.

The third direction addresses the metrological characteristics of weight-based IoT sensing. Load-cell sensors provide direct measurement of mass and are largely independent of lighting conditions and visual disturbances. This makes them suitable for refrigerator-based monitoring systems. However, prior studies indicate that measurement accuracy may be affected by calibration quality, load positioning, and environmental conditions [4]. In particular, positional sensitivity has been identified as a significant factor influencing measurement stability. Additional research highlights the importance of efficient embedded architectures, such as ESP8266-based systems, for long-term IoT deployment [1]. These findings confirm the suitability of weight-based sensing while also emphasizing the need for careful analysis of uncertainty and repeatability.

The fourth research direction concerns the use of robotic process automation in digital workflows. RPA has been widely applied in procurement and supply chain management to automate repetitive tasks and improve process efficiency [2]. It enables interaction with existing web interfaces without requiring dedicated APIs. Further studies demonstrate the potential of RPA for optimizing order-handling processes in supply chains [10]. Despite these advantages, current applications are primarily limited to enterprise environments and do not involve integration with physical sensing systems. As a result, the direct connection between real-time sensor data and automated digital actions remains insufficiently explored.

Recent review studies provide a broader perspective on IoT-enabled inventory and smart kitchen technologies. IoT systems enhance real-time visibility and responsiveness in inventory management [7], while smart kitchen solutions contribute to improved food quality, safety, and sustainability [3]. In addition, developments in online grocery ecosystems highlight the growing role of smart devices in linking household monitoring with digital services [11]. However, these studies are mostly descriptive and do not present experimentally validated closed-loop systems that integrate sensing, communication, and autonomous execution.

Recent works (2023–2025) indicate a transition toward fully autonomous smart home ecosystems that combine sensing, edge computing, and service orchestration. Despite this progress, the integration of low-cost embedded sensing with autonomous digital execution, such as RPA-driven ordering, remains limited, particularly in household-scale applications.

Despite significant advances in IoT monitoring, weight-based sensing, and RPA technologies, existing solutions remain fragmented and lack full integration across physical and digital layers. In particular, there is a lack of experimentally validated systems that combine low-cost sensing with real-time autonomous ordering in a compact household environment. This gap motivates the development of a unified IoT-RPA cyber-physical architecture

capable of closing the loop between measurement, decision-making, and action.

To provide a clearer comparison of existing approaches and to highlight the research gap addressed in this study, Table 1 summarizes key characteristics of representative IoT-based inventory management systems and related solutions.

Table 1. Comparative analysis of IoT-based inventory management systems

Study	System type	Sensing method	Automation level	Main limitations
[1] IoT Inventory System (2023)	Household IoT	Generic sensors	Monitoring + alerts	No autonomous ordering
[2] Smart Kitchen System (2018)	Kitchen IoT	Weight + LDR	Monitoring + app	Requires user interaction
[3] Smart Shelf System (2024)	Retail system	Load cells + vision	Semi-automated	High complexity, cost
[6], [7] RPA in Supply Chain	Enterprise systems	No physical sensing	High (digital only)	No integration with IoT
Proposed system	Household cyber-physical	Load cell (weight-based)	Fully automated	Limited product diversity (current study)

As shown in Table 1, existing systems either focus on sensing and monitoring without autonomous decision execution, or on digital automation without integration with physical sensing. In contrast, the proposed system enables a closed-loop "measurement–decision–action" process by directly linking IoT-based weight sensing with RPA-driven digital ordering. This integration addresses the identified research gap and distinguishes the proposed approach from existing solutions.

2. Research methodology

The research methodology is based on an experimental approach aimed at assessing the performance of an integrated cyber-physical system that combines sensor-based IoT measurement of product weight in a refrigerator with subsequent automated digital data processing using RPA. The research is implemented as a sequential multi-stage process in which the hardware, communication, and software subsystems are tested as a single unit in conditions close to real-world operation.

At the initial stage, the hardware-software system was deployed and configured. The hardware setup consisted of a strain-gauge load cell, a signal amplifier, an HX711 analogue-to-digital converter, and a WeMos D1 (ESP8266) microcontroller equipped with Wi-Fi connectivity. The experiments were carried out in a domestic environment under controlled conditions, with ambient temperature maintained at $20 \pm 1^\circ\text{C}$, stable network connectivity, and a realistic operational workload.

To ensure proper functionality of the sensing chain (load cell-HX711-ESP8266), calibration and preliminary verification measurements were first performed under stable room-temperature conditions. This allowed validation of measurement accuracy and system integrity in a controlled setting. Subsequently, the sensor platform was transferred to a refrigerator environment, where it operated under lower temperatures and natural thermal fluctuations. Such a two-stage approach enabled assessment of system performance both in controlled laboratory-like conditions and in realistic usage scenarios typical for household refrigeration systems.

The software part consists of Arduino firmware for telemetry collection and a local Flask-based server that receives, structures and stores data in CSV format. Before the main experiment, the sensor path is calibrated using a control load with a known mass; the determined calibration coefficient is integrated into the microcontroller program code to correct subsequent measurements. In parallel, the temperature inside the refrigerator

is recorded, since temperature conditions are considered a significant factor affecting the accuracy of the strain gauge. After calibration, a main series of experiments is performed, which includes at least 30 consecutive measurements of the weight of one product.

In the present experimental validation, measurements were conducted using a single representative product type in order to evaluate the stability and functional performance of the proposed IoT sensing subsystem. The objective of the experiment was not to provide a universal metrological characterization for all possible food products, but rather to verify the operational feasibility of the weight-based monitoring architecture. Since strain-gauge sensors measure relative mass variations within a defined operating range, the system behaviour is expected to remain consistent for other products of comparable weight. Nevertheless, different container geometries and mass distributions may influence measurement stability, which will be investigated in future experiments involving multiple product categories.

The selected sample size ($N \geq 30$) is sufficient for preliminary statistical validation, enabling estimation of mean error, variance, and confidence intervals under controlled conditions. The primary goal of this study is to validate the functional feasibility and stability of the integrated IoT-RPA architecture rather than to provide a comprehensive metrological characterization across all product categories. Nevertheless, real-world applications involve heterogeneous products and varying conditions. Future studies will extend the experimental framework to include multiple product types, diverse load distributions, and broader environmental conditions in order to assess system generalizability and robustness.

For each measurement, time stamps, raw sensor readings, corrected weight values, ambient temperature, and data transfer success status are automatically recorded. During the experiment, a controlled reduction in product mass below a preset threshold level is performed, which allows us to assess the system response speed – from the moment a critical weight loss is recorded to the appearance of a corresponding entry in the server CSV file. To check long-term stability, measurements are repeated after 24 hours without recalibration of the sensor, which allows us to assess possible drift of the readings and the reproducibility of the results. Repeatability of measurements was evaluated by analysing multiple consecutive measurements under identical conditions. The low variation between repeated trials confirms the consistency of the sensing subsystem. The second key block of the study is dedicated to testing the automated RPA subsystem implemented in the Microsoft Power Automate Desktop environment. The RPA experiments were conducted using a real consumer-oriented online grocery store operating through a standard web interface. The specific platform name is not disclosed to avoid potential commercial bias and because the automation workflow relies on generic browser interaction mechanisms rather than platform-specific APIs. The implemented RPA scenario interacts with typical e-commerce elements such as product search fields, product pages, and basket management functions that are common across most online grocery platforms. The tested web interface included typical dynamic elements commonly found in modern e-commerce platforms, including asynchronous product loading, dynamic price updates, search filters, and pop-up notification windows. The website structure consisted of multi-level navigation menus, product catalogue pages, and interactive shopping basket components. These characteristics allowed the experiment to realistically simulate the operational conditions encountered by RPA systems interacting with contemporary web-based retail platforms. Consequently, the obtained performance indicators reflect typical robustness challenges associated with UI-driven automation.

Experimental data processing is carried out in the Arduino Serial Monitor and Microsoft Excel environments using basic statistical indicators: mean value, mean absolute error (MAE), standard deviation, coefficient of variation (Cv), fraction of lost

telemetry and correlation of error with temperature. The system efficiency is assessed according to a set of criteria, in particular: MAE no more than 2%, Cv no more than 5%, stable data transmission without systematic losses and the absence of critical failures in the RPA process. The selected statistical indicators are typical for evaluating low-cost sensor systems and household IoT devices, for which the key characteristics are stability, repeatability and relative accuracy of measurements, rather than absolute metrological precision of the laboratory level. Measurement uncertainty was estimated by considering sensor resolution, calibration error, and environmental factors such as temperature variation. The combined uncertainty was approximated using standard error propagation methods, providing an overall uncertainty bound of the measurement system. The methodology takes into account the limitations associated with the sensitivity of strain gauges to temperature fluctuations, potential drift of indicators over time, and the dependence of RPA on changes in the web interface of the online store. Despite these limitations, the proposed approach provides sufficient validity to assess the real-world performance of the system and its practical potential as a prototype for household automated product inventory management.

To improve methodological rigor, additional statistical analysis was performed, including estimation of confidence intervals for the mean measurement error and variance analysis across repeated observations. The 95% confidence interval for the measured weight values was calculated assuming a normal distribution of errors. This allows a more robust interpretation of measurement stability and system reliability.

To ensure reproducibility of the proposed system, the data processing workflow can be described as a sequential algorithm consisting of several stages.

- Step 1: Data acquisition. The ESP8266 microcontroller periodically reads raw values from the HX711 ADC module at a sampling rate of approximately 10 Hz.
- Step 2: Signal filtering and calibration. Raw values are averaged over a short window of 10 measurements to reduce noise. The averaged value is multiplied by a predefined calibration coefficient obtained during the calibration stage.
- Step 3: Threshold analysis. The corrected weight value is compared with a predefined minimum threshold $W < W_{\min}$.
- Step 4: Data transmission. The telemetry packet containing timestamp, raw value, corrected weight, and temperature is transmitted via HTTP POST request to the Flask server.
- Step 5: Server-side storage. The Flask application parses the incoming packet and appends the record to a CSV file used as a shared data interface for the RPA subsystem.
- Step 6: RPA decision logic. The RPA bot periodically scans the CSV file. When a value $W < W_{\min}$ is detected, the bot initiates an automated purchasing workflow via the online store web interface.

In order to formally evaluate the Robotic Process Automation (RPA) subsystem as an integral component of the cyber-physical loop, a set of quantitative performance metrics was defined and experimentally measured. The RPA module was treated as a stochastic execution system operating under real-world web interface constraints.

The following key performance indicators were selected: average RPA execution time (latency), measured from CSV file update to completion of the order simulation; success rate, defined as the ratio of successful executions to total launches; failure classification, grouping execution errors by root cause; mean time between failures (MTBF), calculated as the average number of successful runs between two consecutive failures. From a robustness perspective, the RPA subsystem is sensitive to changes in the web interface structure, dynamic content loading, and security mechanisms such as CAPTCHA. Network

instability may also affect execution consistency. To improve robustness, future implementations may incorporate adaptive element detection, fallback logic, and API-based integration where available. Security considerations include protection of user credentials and safe handling of automated transactions. During the experiment, the RPA scenario was executed $N = 120$ times over a period of 7 consecutive days, under identical functional conditions but with naturally varying network latency and web page response times. Each run was automatically logged, including start time, completion time, execution status, and error type (if applicable).

3. System architecture

The architecture of the proposed system is built as a multi-layer cyber-physical platform that integrates the sensor layer of the Internet of Things, the network layer of wireless data transmission and the application layer of robotic process automation (RPA). The system is designed as a closed loop "measurement–transmission–analysis–action", which minimizes user participation and provides continuous monitoring of the condition of products.

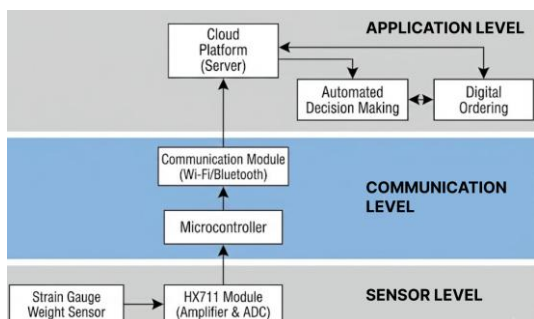


Fig. 1. Functional and structural diagram of the weight sensor system in the refrigerator

As shown in Figure 1, the system architecture is structurally divided into three interconnected levels: the sensor level is responsible for weight measurement; the microcontroller and communication level provides primary processing and transmission of telemetry; the application level (RPA) implements automated decision-making and digital ordering.

The lower level of the architecture forms a sensor module, which is based on a strain-resistive weight sensor mechanically fixed on a platform inside the refrigerator. Under the action of the product mass, the sensor deforms, which is converted into a change in the electrical signal according to the principle of the strain bridge.

Unlike typical kitchen-scale implementations described in previous studies, the proposed prototype integrates the load cell within a rigid support platform specifically adapted for refrigerator shelf placement. This mechanical integration improves measurement stability and reduces sensitivity to product displacement during normal refrigerator use.

To increase the sensitivity, stability and accuracy of measurements, the signal from the sensor passes through the HX711 module, which performs the functions of amplification and analogue-to-digital conversion (ADC). The use of HX711 provides high measurement resolution, low noise and data suitability for further digital processing at the microcontroller level. In order to correctly integrate the sensor node into the system, a half-bridge configuration of the load cell connection was used, which provides a balance between accuracy, stability and hardware complexity. The diagrams shown in Figure 2 and Figure 3 demonstrate the electrical integration of the load cell with the HX711 module, which is a key part of the system's sensor path. This configuration allows you to minimize the impact of electrical interference and increase the reproducibility of measurement results.

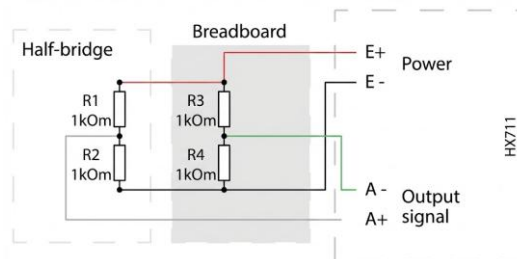


Fig. 2. Connection diagram of a half-bridge strain gauge to the HX711

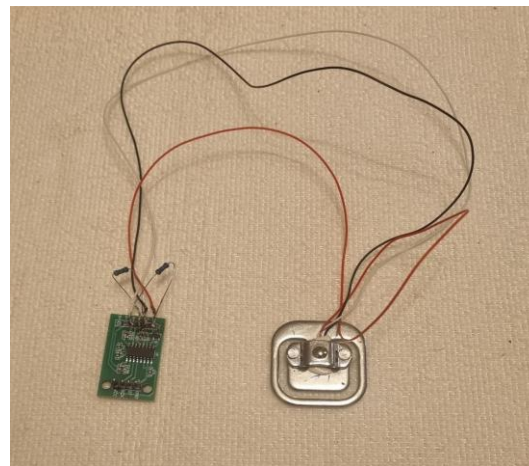


Fig. 3. Connecting a half-bridge strain gauge to the HX711 module

The intermediate layer of the system architecture is realized using a WeMos D1 microcontroller based on the ESP8266 platform, which functions as the core unit for data acquisition, preprocessing, and wireless telemetry transmission. The microcontroller periodically retrieves measurements from the HX711 module, applies a predefined calibration coefficient, and generates a structured data packet that includes a timestamp, raw sensor readings, calibrated weight values, and the internal temperature of the refrigerator. Wireless communication is enabled through the integrated ESP8266 Wi-Fi interface, allowing seamless data transfer to a local server without requiring additional networking hardware.

Figures 4–7 illustrate the configuration of the development environment and the integration of the necessary software libraries that support firmware implementation and network communication for the ESP8266-based sensing node within the experimental setup.

The communication layer of the system is implemented on a local server built using the Flask microframework. The server receives telemetry data from the WeMos D1 microcontroller via HTTP POST requests, performs their basic structuring and sequentially accumulates it in a CSV format file. This approach ensures the preservation of historical measurements, their availability for further analytics and compatibility with the RPA platform. In addition, the server performs the function of an intermediate data buffer, which increases the system's resistance to short-term network failures.

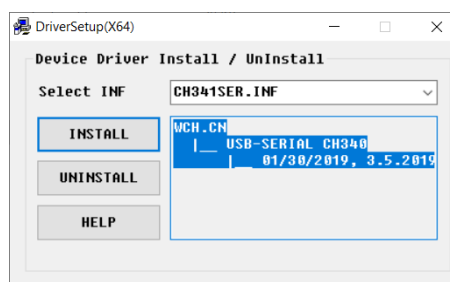


Fig. 4. CH340 driver installation menu

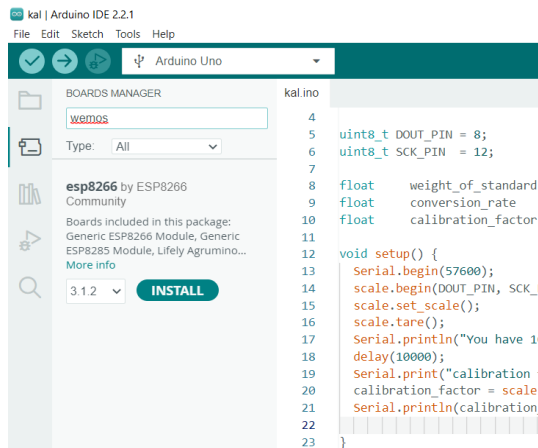


Fig. 5. Finding and installing the esp8266 library

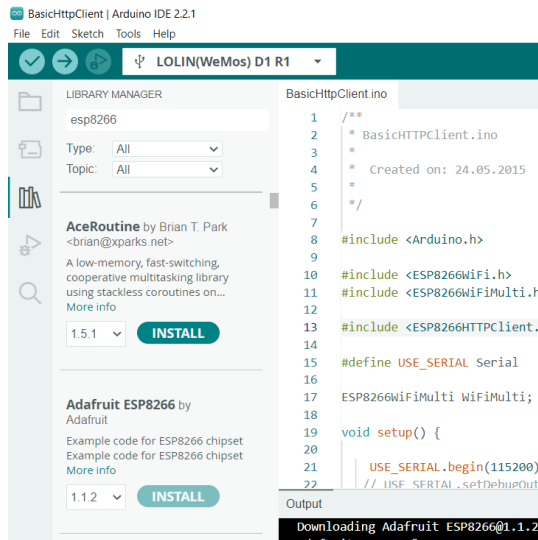


Fig. 6. Finding and installing the adafruit ESP8266 library

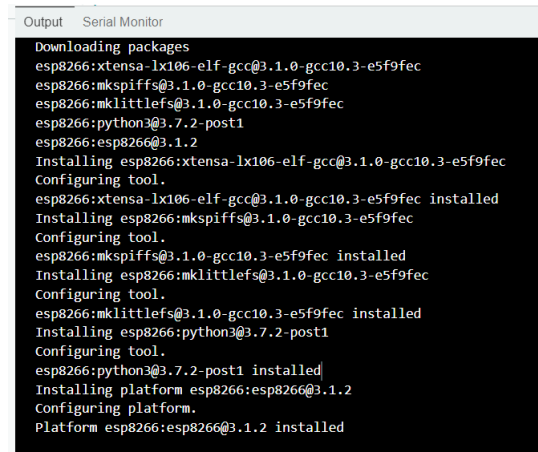


Fig. 7. Library installation process in Arduino IDE

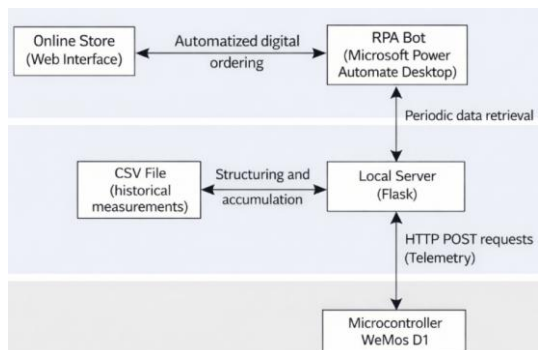


Fig. 8. Data transfer scheme: WeMos D1 → Flask → CSV → RPA

As Figure 8 illustrates, the upper (application) layer of the architecture is implemented using robotic process automation in the Microsoft Power Automate Desktop environment. Although Microsoft Power Automate Desktop was selected for the prototype implementation due to its accessibility and integration with Windows environments, similar automation workflows could also be implemented using open-source or community RPA platforms such as UiPath Community Edition or Robot Framework. These alternatives provide comparable browser automation capabilities and could support the same closed-loop architecture. The choice of Power Automate in this study was primarily motivated by its rapid prototyping capabilities and straightforward integration with local file-based data sources such as CSV logs generated by the Flask server.

The RPA bot periodically accesses the updated CSV file, reads the latest weight values, and compares them with a predefined minimum threshold. In case of a critical weight loss, the system automatically initiates a digital order scenario: the bot opens the web interface of the online store, searches for the corresponding product, checks its availability and price, and simulates the order process.

This approach provides a fully automated inventory management cycle without the need for constant user intervention. The interaction between the architecture levels is organized in the form of a closed loop: the physical weight loss of the product is recorded by a strain gauge, digitized via HX711, processed by WeMos D1, transmitted to the Flask server, where it is structured and stored, after which the RPA subsystem responds with an automated action. This scheme ensures the continuity of inventory control and allows the system to function autonomously for a long time. From a performance perspective, the system latency is primarily determined by sensor sampling rate, network transmission time, and RPA execution delay. Experimental results show that the end-to-end response time remains within practical limits for household applications. In terms of scalability, the architecture supports multiple sensor nodes; however, increased system load may lead to higher server processing time and RPA queue delays. Future implementations may require asynchronous processing and distributed architectures to maintain performance under higher loads.

In terms of scalability, the architecture provides the ability to connect multiple sensor nodes to monitor different products simultaneously. Each node can send data to the same server, and the RPA module can process it in batch or event mode. In the future, the system can be expanded by using an ESP32 microcontroller, implementing the MQTT protocol for energy-efficient telemetry, or integrating machine learning-based consumption forecasting algorithms.

Thus, the proposed architecture combines sensor, network, and software components into a single integrated system capable of not only registering changes in product inventory, but also autonomously responding to them through digital channels of interaction with online stores. This makes the system suitable as a prototype of a new generation smart refrigerator and creates a basis for further research in the field of home automation and intelligent inventory management.

To formalize and visualize the functional interactions within the proposed system, a UML use case diagram was developed, representing three primary actors: the User, the Automated System, and the Online Store [11]. The diagram defines the system boundaries and illustrates the division of responsibilities between human actions and automated processes.

The user is responsible for the initial configuration of the system, including sensor calibration, network setup, product parameter definition, and authorization in the online store. After configuration, the system operates autonomously, periodically measuring product weight, updating inventory data, and identifying shortages based on predefined threshold values.

When a product deficit is detected, the automated system generates a purchase list and initiates interaction with the online store through an RPA-driven workflow. The RPA component performs product search, price verification against user-defined

constraints, and basket management. Final actions such as order confirmation and payment remain under user control, ensuring security and compliance with transactional requirements. The UML use case diagram provides a high-level representation

of the system logic and interaction flow, supporting the architectural design and facilitating the structured implementation of RPA processes without overloading the article with low-level procedural details (Figure 9).

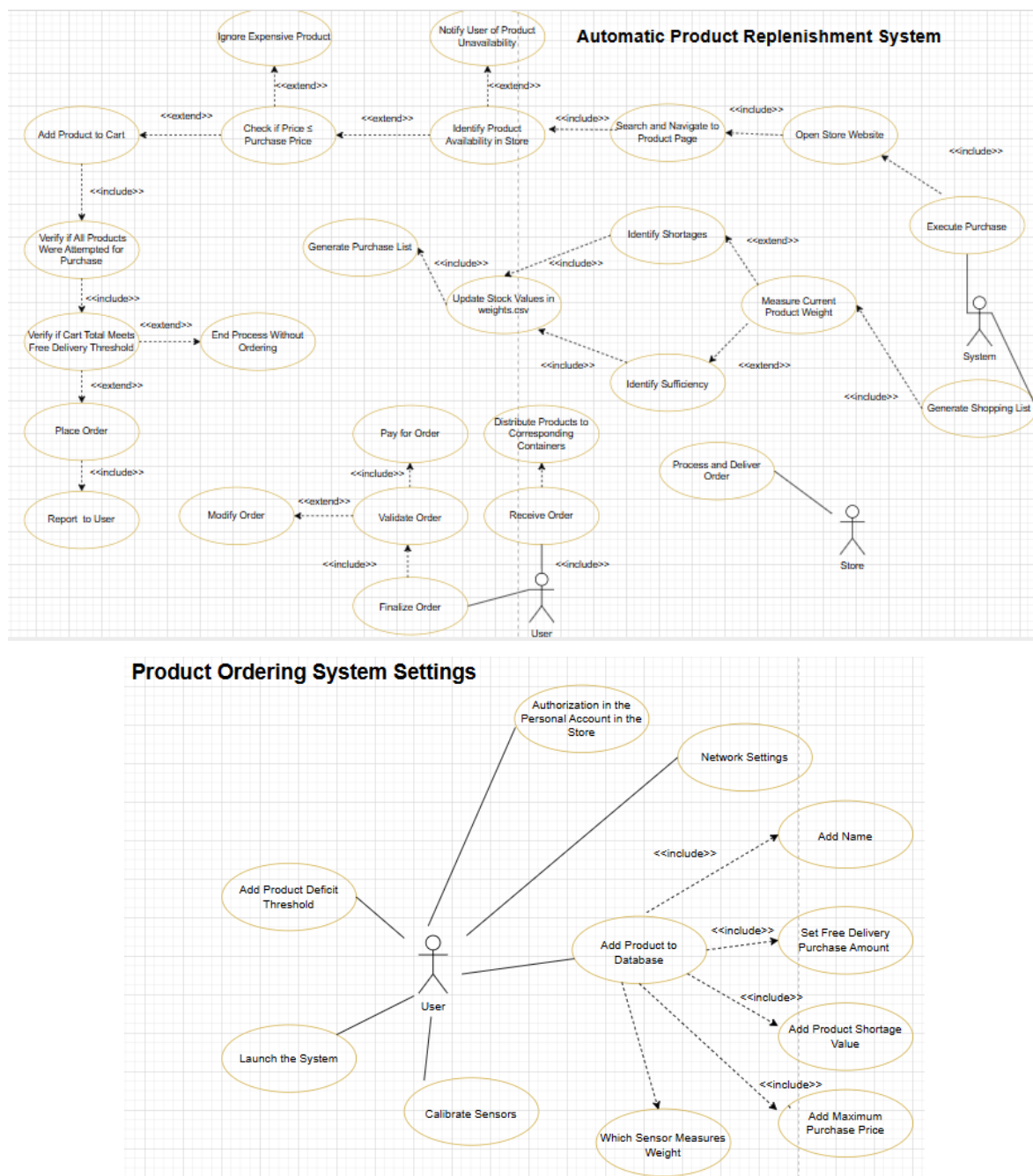


Fig. 9. Automated system (UML use case diagram)

4. Results

A load cell-based weight measurement system requires careful planning and selection of appropriate solutions. When choosing a microcontroller for a prototype, it is important to determine the needs and the price-performance ratio, as the market offers many options, from simple to quite complex.

Initial results were obtained during a control measurement of the weight of an object with a known mass. At this stage, the correct operation of the strain gauge, HX711 amplifier and WeMos D1 microcontroller as part of a single system was verified. Visual confirmation of the procedure is shown in the Figure 10.

The obtained values demonstrated the stability of the signal at room temperature, which allowed us to proceed to the calibration procedure. Calibration was carried out by repeatedly weighing the control weight and correcting the calibration

coefficient in the microcontroller program code. The corresponding process is illustrated in the Figure 11.

After calibration, the system was placed in a refrigerator, where the ambient temperature was approximately 20.1°C at the time of the initial measurement, and then varied according to the operating mode of the cooling device. This allowed us to assess the effect of temperature on the stability of the strain gauge readings.

Additional observations during the experiment indicated that sudden temperature fluctuations inside the refrigerator did not cause critical measurement instability. The HX711 module and strain-gauge sensor exhibited gradual drift rather than abrupt signal jumps. Short-term temperature changes within the typical refrigerator operating range (approximately 3–8°C) resulted in deviations that remained within the acceptable measurement error margin.

Mechanical disturbances such as short-term vibrations or accidental product displacement produced temporary measurement spikes; however, these anomalies were effectively suppressed by the averaging procedure implemented in the microcontroller firmware. As a result, the system demonstrated stable operation under typical domestic usage conditions.



Fig. 10. Control measurement of the object weight

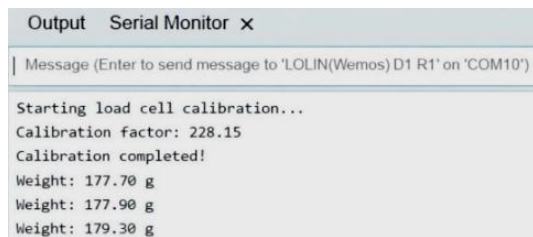


Fig. 11. Calibration and measurement results

Although strain-gauge sensors exhibit temperature-dependent sensitivity, the observed deviations remained within acceptable limits for household monitoring applications. In practical implementations, such deviations may be further reduced using software-based temperature compensation techniques or periodic recalibration. To quantitatively evaluate the relationship between temperature variation and measurement accuracy, a Pearson correlation analysis was performed between recorded temperature values and the corresponding measurement errors. The calculated correlation coefficient was $r = 0.18$, indicating a weak positive correlation between temperature fluctuations and measurement deviation.

This result suggests that, within the observed refrigerator operating range, temperature variations have only a minor influence on measurement accuracy. The magnitude of the temperature-induced error remained significantly below the predefined acceptance threshold of 2%, confirming the suitability of the sensing subsystem for household monitoring applications.

The system successfully recorded a critical weight loss and promptly generated a corresponding entry in a CSV file on the server. The reaction time between the physical weight loss and the digital registration of the event was small enough for practical use in real refrigerator use.

To assess long-term stability, the experiment was repeated after 24 hours without recalibration of the sensor. Comparison of the results showed the absence of significant drift of the indicators, which indicates the reliability of the combination of HX711 and WeMos D1 in long-term operation. Minor deviations were explained by temperature fluctuations and mechanical micro-displacements of the product on the sensor platform. The main sources of measurement error include temperature-dependent drift of the strain gauge, non-uniform load distribution, and external mechanical disturbances. Additionally, electrical noise in the HX711 module and Wi-Fi transmission delays may introduce minor variability. These factors collectively contribute to the observed measurement uncertainty.

Previous studies indicate that the measurement accuracy of low-cost load cells may depend on load placement across the sensor platform [4]. In the proposed prototype, the load cell was mounted beneath a rigid support platform designed to promote approximately central placement of the product during normal refrigerator use. A dedicated positional sensitivity exper-

iment was not included in the present study because the main objective was to validate the integrated IoT-RPA system architecture rather than to perform a detailed metrological characterization of the sensor. Nevertheless, systematic evaluation of load distribution effects will be included in future experimental work.

Quantitative evaluation of the RPA subsystem demonstrated stable operational behaviour under real web conditions. The average execution time of the RPA scenario was 27.4 s, with a standard deviation of ± 4.8 s. The minimum observed execution time was 19.2 s, while the maximum reached 41.6 s, mainly due to variable page loading times.

Out of 120 RPA executions, 112 runs were completed successfully, resulting in a success rate of 93.3%. The remaining 8 failures (6.7%) were analysed and classified according to their root causes, as summarized below: UI structure changes of the online store website – 3 cases (37.5% of failures); temporary network delays or timeouts – 3 cases (37.5%); unexpected pop-up dialogs or CAPTCHA elements – 2 cases (25%). Based on the observed execution sequence, the mean time between failures (MTBF) was estimated as 14.0 successful runs, which confirms acceptable robustness of the RPA module for non-critical household automation tasks. Importantly, no failures resulted in system deadlock or data corruption. After each unsuccessful execution, the RPA scenario was able to restart correctly without manual intervention, indicating functional fault tolerance of the proposed automation logic.

Regarding data transmission, the system demonstrated high reliability: more than 10,000 telemetry records were successfully transmitted and stored without loss of information. This indicates the stability of the WeMos D1 Wi-Fi connection and the correct operation of the Flask-based server part. The integration of IoT sensors with RPA creates an effective closed loop of automated inventory management, which can be scaled to a larger number of products or integrated into commercial smart refrigerators. At the same time, further research can be aimed at using more powerful microcontrollers (e.g., ESP32), implementing algorithms for predicting product consumption, and developing a specialized mobile application for real-time balance monitoring.

To provide a clearer overview of the experimental evaluation, the main performance indicators of both the IoT sensing subsystem and the RPA automation module are summarized in Table 2. The results are compared with the predefined acceptance criteria used in this study.

Table 2. Summary of experimental performance indicators of the proposed IoT-RPA system

Subsystem	Performance indicator	Experimental value	Acceptance criterion	Status
IoT sensing	Mean Absolute Error (MAE)	< 2%	$\leq 2\%$	Met
IoT sensing	Coefficient of Variation	< 5%	$\leq 5\%$	Met
IoT sensing	Long-term drift (24 h)	< 0.5%	$\leq 1\%$	Met
RPA automation	Average execution time	27.4 s $\pm$ 4.8 s	-	-
RPA automation	Minimum execution time	19.2 s	-	-
RPA automation	Maximum execution time	41.6 s	-	-
RPA automation	Execution success rate	93.3%	$\geq 90\%$	Met
RPA automation	Mean time between failures	14 runs	-	-

Compared to similar IoT-based inventory monitoring systems reported in [6, 8], which primarily focus on data acquisition and user notification, the proposed system demonstrates not only comparable measurement accuracy (MAE < 2%) but also extends functionality toward fully autonomous operation. In particular, while existing household IoT solutions typically lack automated decision execution, the integration of RPA in the proposed architecture enables direct transition from sensing to action without user intervention.

In contrast to more complex retail-oriented systems such as [5], which combine load cells with computer vision and require higher computational and infrastructural resources, the proposed solution achieves reliable performance using a low-cost embedded sensing approach. Additionally, the RPA subsystem demonstrated a success rate of 93.3% with an average execution time of 27.4 s, indicating that practical end-to-end automation can be achieved without dedicated API integration. These results confirm that the proposed system provides a favourable balance between accuracy, system complexity, and level of automation compared to existing approaches.

5. Conclusions

Experimental evaluation confirmed that the proposed system provides reliable measurement performance and stable operation, while the RPA subsystem demonstrated consistent execution under real-world conditions. The RPA subsystem achieved a 93.3% execution success rate across 120 automated runs, with an average execution time of 27.4 s and an estimated mean time between failures of 14 runs.

The results confirm that the integration of low-cost IoT sensing with robotic process automation enables the creation of a practical cyber-physical system for autonomous household inventory monitoring and replenishment. The proposed architecture combines weight-based sensing, Wi-Fi telemetry transmission via ESP8266, server-side processing using Flask, and automated ordering implemented through Microsoft Power Automate Desktop.

Despite the promising results, several limitations of the present study should be clearly acknowledged. First, the experimental validation was conducted using a single product type and a single sensor configuration, which limits the generalizability of the results to a wider range of food products with different geometries and mass distributions. Second, the experimental duration was relatively short, and long-term system behaviour under continuous operation was not systematically evaluated. Third, the study did not include large-scale validation with multiple sensor nodes operating simultaneously, which may affect system performance under increased load. In addition, the RPA subsystem depends on the stability of external web interfaces, making it sensitive to UI changes, dynamic content, and security mechanisms such as CAPTCHA. From a practical deployment perspective, challenges may arise in integrating the system with commercial online platforms, ensuring secure user authentication, and maintaining long-term operational reliability. These limitations indicate the need for further research focused on scalability, robustness, and real-world deployment scenarios.

Future work will focus on extending the system through the use of more powerful microcontrollers such as ESP32, implementing MQTT-based communication for improved energy efficiency, integrating food consumption forecasting algorithms, and increasing the robustness of the RPA subsystem to dynamic web-interface changes.

Overall, the proposed IoT-RPA architecture demonstrates the feasibility of closing the loop between physical inventory sensing and automated digital ordering, contributing to the development of intelligent household cyber-physical systems.

References

- Akintade, O. O., Yesufu, T. K., & Kehinde, L. O. (2019). Development of Power Consumption Models for ESP8266-Enabled Low-Cost IoT Monitoring Nodes. *Advances in Internet of Things*, 09(01), 1–14. <https://doi.org/10.4236/ait.2019.91001>
- Flehsig, C., Anslinger, F., & Lasch, R. (2022). Robotic Process Automation in purchasing and supply management: A multiple case study on potentials, barriers, and implementation. *Journal of Purchasing and Supply Management*, 28(1), 100718. <https://doi.org/10.1016/j.pursup.2021.100718>
- Golshany, H., Ni, Y., Yu, Q., & Fan, L. (2025). IoT-enabled smart kitchen technologies and their impact on food storage, preparation, and culinary experiences: A systematic review. *Food Research International*, 213, 116557. <https://doi.org/10.1016/j.foodres.2025.116557>
- Habiba, C., & Prayogo, A. (2025). Characterizing Positional Sensitivity of Low-Cost Load Cells for IoT-Based Weighing Systems. *Internet of Things and Artificial Intelligence Journal*, 5(3), 778–796. <https://doi.org/10.31763/iota.v5i3.1002>
- Jose, J. A. C., Bertumen, C. J. B., Roque, M. T. C., Umali, A. E. B., Villanueva, J. C. T., TanAi, R. J., Sybingco, E., San Juan, J., & Gonzales, E. C. (2024). Smart Shelf System for Customer Behavior Tracking in Supermarkets. *Sensors*, 24(2), 367. <https://doi.org/10.3390/s24020367>
- Kiruthika, J. K., Manikandan, S., Nithya, U., Preethika, M., & Poorna, S. (2023). IoT-Based Smart Inventory Management System. *2023 International Conference on Computer Communication and Informatics (ICCCI)*, 1–6. <https://doi.org/10.1109/ICCCI56745.2023.10128211>
- Mashayekhy, Y., Babaei, A., Yuan, X.-M., & Xue, A. (2022). Impact of Internet of Things (IoT) on Inventory Management: A Literature Survey. *Logistics*, 6(2), 33. <https://doi.org/10.3390/logistics6020033>
- Rezwani, S., Ahmed, W., Mahia, M. A., & Islam, M. R. (2018). IoT Based Smart Inventory Management System for Kitchen Using Weight Sensors, LDR, LED, Arduino Mega and NodeMCU (ESP8266) Wi-Fi Module with Website and App. *2018 Fourth International Conference on Advances in Computing, Communication and Automation (ICACCA)*, 1–6. <https://doi.org/10.1109/ICACCA.2018.8776761>
- Samchuk, L., & Povstiana, Y. (2024). UML diagrams of the management system of maintenance stations. *Informatyka, Automatyka, Pomiar w Gospodarce i Ochronie Środowiska*, 14(4), 141–145. <https://doi.org/10.35784/iapgos.6320>
- Shamsuzzoha, A., & Pelkonen, S. (2025). A robotic process automation model for order-handling optimization in supply chain management. *Supply Chain Analytics*, 9, 100102. <https://doi.org/10.1016/j.sca.2025.100102>
- Stecula, K., Wolniak, R., & Aydın, B. (2024). Technology Development in Online Grocery Shopping—From Shopping Services to Virtual Reality, Metaverse, and Smart Devices: A Review. *Foods*, 13(23), 3959. <https://doi.org/10.3390/foods13233959>

Ph.D. Lyudmila Samchuk

e-mail: samchuk204@gmail.com

Candidate of Technical Sciences, associate professor, Faculty of Transport and Mechanical Engineering, Department of Applied Mechanics and Mechatronics, Lutsk National Technical University. Scientific interests include the technology for manufacturing porous permeable materials in SHS mode. Utilization of grinding sludge from bearing production. Software tools for visualizing the results of SHS experiments and structural parameters of materials. Interactive visualization of data from research on porosity and permeability of materials

<https://orcid.org/0000-0003-2516-045X>

Ph.D. Yuliia Povstiana

e-mail: yuliapovstiana@ukr.net

Candidate of Technical Sciences, associate professor, Faculty of Computer and Information Technologies, Department of Software Engineering, Lutsk National Technical University. Scientific interests include soft skills, digital and communication technologies in education, computer graphics and image processing, and analysis using UML diagrams. Author of more than 80 scientific publications.

<https://orcid.org/0000-0001-5426-4157>

Ph.D. Nataliia Lishchyna

e-mail: lishchyna@gmail.com

Associate professor, Head of the Department of Software Engineering, Lutsk National Technical University. Author of more than 70 scientific publications. Research interests: software development, data mining, innovative technologies in education.

<https://orcid.org/0000-0002-5200-536X>

Ponomarenko Mykyta

e-mail: mykyta.ponomarenko@gmail.com

Master's in computer and information technologies at Lutsk National Technical University.

<https://orcid.org/0009-0001-3236-7515>

