REPUBLIC OF RWANDA

ULK POLYTECHNIC INSTITUTE

P.O Box 2280

Website://www.ulkpolytechnic.ac.rw

E-mail: polytechnic.institute@ulk.ac.rw

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

OPTION: ELECTRONICS AND TELECOMMUNICATION TECHNOLOGY

ACADEMIC YEAR 2023/2024

FINAL YEAR PROJECT

DESIGN AND IMPLEMENTATION OF A WEIGHT SENSOR ALARM CLOCK USING AN ARDUINO UNO

Research project submitted in partial fulfillment of the requirement for award of advanced diploma in Electronics and Telecommunication Technology

Submitted by: AMULI MUZUKA Christopher

Roll number: 202150125

SUPERVISOR: Ir. KARIKURUBU EMMANUEL

September 2024

DECLARATION A

I, Amuli Muzuka Christopher, hereby declare that this research work is my original work and has not previously been submitted for any degree or other academic award in any university or educational institution. No part of this research may be reproduced without my consent or that of ULK Polytechnic Institute.

Student name : AMULI MUZUKA Christopher

Signature: ....

Date: ....

DECLARATION B

I confirm that the work presented in this research project was conducted by the candidate under my supervision and is being submitted with my approval as the UPI supervisor.

Name of supervisor: Ir. KARIKURUBU Emmanuel

Signature: ....

Date: ....

DEDICATION

I dedicate this final year of study to Almighty God, whose unwavering guidance and blessings have enabled me to reach this milestone.

To my very dear parents, CHANVU Muzuka Frédéric and BYENDA Kilolo Irène, your unfailing support and faith in me have made this achievement possible.

To all my brothers and sisters, your encouragement and complicity have been invaluable throughout this journey.

To my esteemed supervisor, KARIKURUBU Emmanuel, your advice and patience have been crucial to my academic development.

To all my teachers, your dedication to teaching has greatly enriched my educational experience.

To my colleagues and friends, your support and guidance have been essential.

This success is a reflection of the collective love, support and effort you have all given me during

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to all those who have played an important role in the completion of this research project.

First of all, I thank Almighty God for his guidance, blessings and strength throughout this journey.

I am deeply grateful to my parents, CHANVU Muzuka Frédéric and BYENDA Kilolo Irène, for their constant encouragement, their faith in my potential and the sacrifices they have made to ensure my success.

Special thanks go to my supervisor, KARIKURUBU Emmanuel, for his expertise, advice and patience, which have played a decisive role in my academic development.

I also express my gratitude to my instructors, whose dedication to teaching made learning an enjoyable experience.

To my colleagues and friends, your support, cooperation and camaraderie have been essential throughout this research process.

Finally, to my brothers and sisters, your support and guidance throughout this academic journey has been invaluable.

This research project has been a success, largely thanks to the collective love and support of each and every one of you. I am deeply indebted to each and every one of you for your presence in my life. Thank you for helping me meet this challenge.

ABSTRACT

In today's world, procrastination and difficulty getting out of bed are common challenges for many people. This project tackles these problems by developing an alarm system based on a weight sensor using Arduino Uno. The system aims to improve morning motivation and reduce laziness by ensuring that users have to physically get out of bed to deactivate the alarm.

The design incorporates a weight sensor and an Arduino Uno microcontroller, a keyboard for entering weight parameters, and the other components we'll look at below. To deactivate the alarm, the user has to stand up and place their foot on the sensor, which detects their weight. The alarm only stops if the weight detected is within ±2.5% of the pre-set value, which ensures that the user has actually got out of bed and not used an object to fool the system.

The prototype features a buzzer to trigger the alarm, an LCD screen to display relevant information and an RTC timer for precise alarm management, all housed on a printed circuit board for durability. By applying this physical requirement and focusing on accurate weight detection, the project aims to provide an effective solution to morning procrastination, helping users to start their day more actively.

TABLE OF CONTENTS

TABLE OF FIGURES

LIST OF ABREVIATIONS

AI : Artificial Intelligence

DS3231 : A specific type of Real-Time Clock

FRW : Rwandan Francs

HX711 : A specific type of load cell amplifier

I2C : Inter-Integrated Circuit

I/O : Input/Output

IoT : Internet of Things

kg : Kilogram

LCD : Liquid Crystal Display

PCB : Printed Circuit Board

Pcs : Pieces

PWM : Pulse Width Modulation

RTC : Real-Time Clock

CHAPTER ONE : GENERAL INTRODUCTION

1.1 INTRODUCTION

Waking up on time is a common problem for many people, often leading to delays and reduced productivity. As technology advances, innovative solutions to everyday challenges become increasingly important. One such challenge is effectively motivating people to get out of bed and actively start their day. Traditional alarms can be easily ignored, leading to procrastination and inefficiency.

This study addresses the need for an effective wake-up solution by designing a weight sensor-based alarm system using Arduino Uno. The system ensures that users have to physically get out of bed and step on a weight sensor to turn off the alarm, effectively combating morning procrastination.

The Arduino-based weight sensor alarm system integrates various components to provide a complete solution and by incorporating these components, the system ensures that the alarm only stops when the weight detected on the sensor is within ±50g the user's preset weight. This mechanism prevents users from placing objects on the sensor to avoid getting out of bed, this promoting a more active and engaging start to the day.

1.2 BACKGROUND OF THE STUDY

Manual wake-up solutions often lack efficiency, leading to problems such as repeated use of the snooze button and increased morning procrastination. Alarm automation offers a solution by incorporating technology to reinforce physical engagement. Arduino micro-controllers offer the flexibility needed for such innovations, and weight sensors guarantee accurate measurement. LCD screens and keyboards enhance user interaction, making the system more user-friendly and efficient. This study aims to design an Arduino-based alarm system that improves morning routines and promotes productivity.

1.3 STATMENT OF THE PROBLEM

The problem addressed in this project is the ineffectiveness of traditional alarm systems in motivating users to get out of bed. Many existing alarms are easily ignored, leading to prolonged procrastination and reduced productivity. This project proposes an automated weight-sensor alarm system that requires users to physically stand on the alarm to deactivate it. This approach aims to improve efficiency, reduce procrastination and boost user motivation to start the day actively.

1.4 RESEARCH OBJECTIVES

The aim of this project is to develop an alarm system based on an Arduino weight sensor to combat morning procrastination and improve wake-up efficiency. By automating the wake-up process and ensuring that the alarm only goes off when the user meets specific weight conditions, the system aims to improve morning productivity and motivation.

1.4.1 General objective

Develop an automated alarm system using an Arduino Uno, LCD screen, and a weight sensor to enhance morning routines, minimize human intervention, and encourage active engagement to turn off the alarm.

1.4.2 Specific objectives

i. To esign the hardware configuration: Configure the Arduino Uno, LCD screen, weight sensor and other components in a functional configuration for accurate weight detection.

ii. To evelop Arduino code: Write and upload Arduino code to interface with the weight sensor, and manage alarm activation based on weight measurements.

iii. To system calibration: Perform calibration tests to guarantee the accuracy of weight measurements and fine-tune alarm activation thresholds.

iv. To implement user interface: Program the LCD screen to provide clear, real-time information on alarm status and weight parameters.

v. To ensure reliability: Test the system to ensure that it operates reliably under different conditions and scenarios.

1.5 Research questions / hypothesis

Here are some research questions and hypotheses concerning the weight sensor alarm system:

Research Questions:

1) How can the reliability and effectiveness of weight-sensor alarm systems be optimized to motivate users to get out of bed?

2) How can the system be designed so that users have to physically engage to deactivate the alarm, thus reducing the likelihood of false alarms?

3) What are the advantages and limitations of using weight sensors in alarm systems compared with traditional wake-up methods?

4) How can the user interface be improved to provide clear feedback and enhance user interaction with the alarm system?

5) How can the weight sensor calibration process be simplified to ensure accurate operation across different users?

1.6 Scope and limitation

This project focuses on the development of an innovative weight sensor alarm clock using an Arduino Uno, aimed at improving morning routines by ensuring that users physically get out of bed. It involves the design and integration of a weight sensor, buzzer and LCD screen to create a functional, user-friendly system. The project is intended for personal use in various residential environments, and its development runs from July 1, 2024 to September 9, 2024. However, there are limitations such as budgetary constraints that may impact on the range of functionality, technological dependencies on the reliability of the Arduino Uno and sensor, and variability in efficiency depending on the user's individual weight and sleeping habits. Despite these challenges, the project aims to provide a practical solution for improving wake-up routines and reducing procrastination.

1.7 SIGNIFICANCE OF THE STUDY

The weight sensor alarm system addresses the common problem of morning procrastination by ensuring that users have to physically engage with the system to deactivate the alarm. This innovation reduces wasted time by programming multiple alarms to delay some and improves productivity by encouraging a more active and engaging waking routine.

1.8 ORGANIZATION OF THE STUDY

The research work is divided into five chapters, as shown below:

Chapter 1: General introduction

This chapter introduces the project, including background, problem statement, research objectives, research questions, scope and limitations. It defines the background and importance of the weight sensor alarm system and outlines the objectives of the study.

1. **Chapter 2:** Literature review

This chapter defines key terms related to the project and reviews existing systems and technologies similar to the weight sensor alarm system. It explores how previous research has addressed the challenges associated with alarm systems, weight sensors and automation, laying the groundwork for the study's contributions.

2. **Chapter 3:** Research methodology

This chapter details the research methods used to design and implement the weight sensor alarm system. It includes an analysis of the system requirements, design methodologies and data collection techniques used to develop and test the system.

3. **Chapter Four:** System Design, Analysis and Implementation

This chapter describes the design and implementation of the weight sensor alarm system. It includes technical details of the hardware and software components, such as the Arduino Uno, the weight sensor, the LCD screen and other integrated elements. Screenshots and diagrams are provided to illustrate the functionality and user interface of the system.

4. **Chapter Five:** Conclusion and Recommendations

This chapter summarizes the findings of the research, highlighting the effectiveness of the weight sensor alarm system in improving waking routines and reducing procrastination. It presents conclusions based on the results of the study and makes recommendations for future improvements and potential areas of research.

CHAPTER 2: LITERATURE REVIEW

2.1. Introduction

A review of the literature on weight-sensing alarm clocks reveals a diverse field of research focused on improving alarm clock systems. The studies highlight the importance of effective wake-up mechanisms, particularly those that address common problems such as sleepiness and procrastination. Research is exploring various sensor technologies, including weight sensors, pressure sensors and motion detectors, to create more reliable and attractive wake-up systems. Researchers are investigating different signaling methods, such as alarms, vibrations and visual notifications, to ensure that users respond to the wake-up call. Key challenges, such as sensor accuracy, user engagement and cost-effectiveness, are discussed, highlighting innovative solutions to improve wake-up efficiency and reduce morning procrastination.

2.2 Specifications

A. Arduino UNO

Arduino Uno The Arduino Uno is equipped with an ATmega328P microcontroller clocked at 16 MHz, 14 digital I/O pins (including 6 for PWM) and 6 analog inputs. It can be powered via USB or an external power supply (7-12V). The board is programmed using the Arduino IDE, making it a versatile choice for prototyping and interactive projects. The Arduino Uno acts as the central controller, processing the weight sensor inputs, managing the outputs to the buzzer and LCD screen, and handling user interactions.

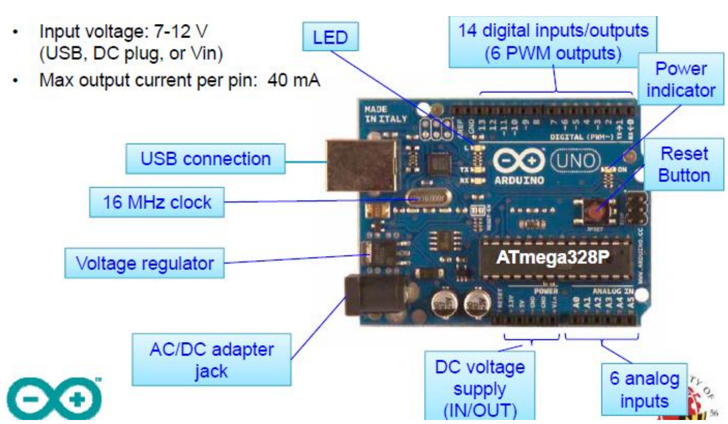


Figure 1 Arduino Uno

B. Weight Sensor (10 kg)

The weight sensor, or load cell, is designed to measure weight with a capacity of up to 10 kg. It operates on the principle of strain gauges, which detect changes in resistance as weight is applied. The sensor produces an analog signal proportional to the weight, which is then converted into a digital signal by the Arduino Uno. This signal is used to determine if the user has gotten out of bed and to activate the alarm accordingly.

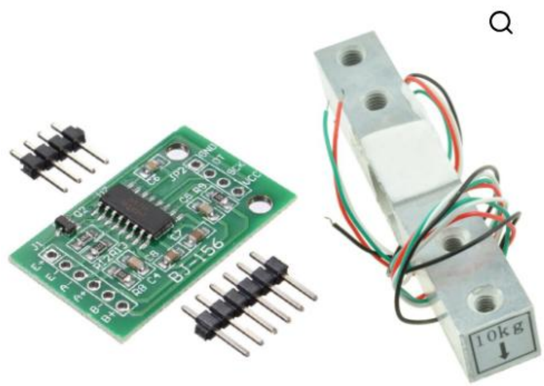


Figure 2 Weight sensor

C. Liquid Crystal Display (LCD)

The LCD display is a 16x2 alphanumeric screen, capable of displaying 16 characters per line on two lines. It operates in 8-bit or 4-bit mode and is used to display weight readings, system status and messages. The display is connected via an I2C module, simplifying wiring and communication with the Arduino Uno.

Figure 3 LCD display

D. I2c module

The I2C module is used to interface the LCD display with the Arduino Uno. It provides a more efficient and rational connection than traditional parallel interfaces, reducing the number of wires required. The module communicates using the I2C protocol, enabling easy integration of the LCD display into the project.

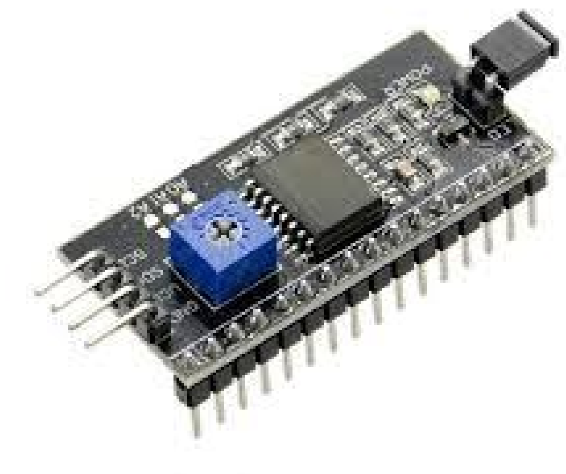


Figure 4 L2C module

E. RTC Timer (Real-Time Clock)

The RTC Timer, like the DS3231, maintains precise time with minimal drift. It operates on the I2C protocol and is used to keep track of the current time and program alarms. The RTC is essential for recording time events and ensuring that the alarm operates according to the user's settings.

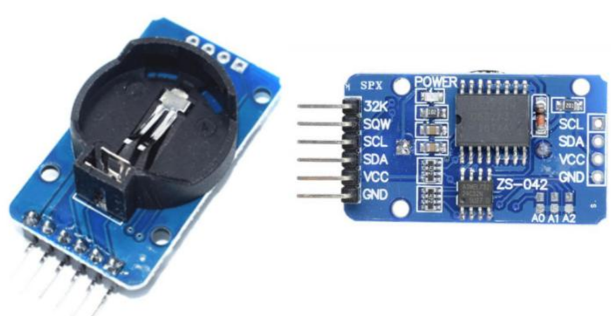


Figure 5 RTC Timer

F. Keyboard (4x4 matrix)

The keyboard is a 4x4 matrix with 16 buttons, allowing the user to enter parameters and passwords. Each press of a button sends a signal to the Arduino Uno, enabling the user to interact, for example, by setting the alarm time or adjusting system parameters. The keyboard is connected to the Arduino's digital pins.

Figure 6 Keypad matrix 4x4

G. Buzzer

The buzzer is a piezoelectric device that emits an audible sound when activated. It is used to alert the user when the weight sensor detects the user's weight, or to signal other important events. The buzzer is powered by the Arduino Uno and produces a loud sound to ensure that the alarm is heard.

Figure 7 Buzzer

Connecting wires are flexible, insulated wires used to create temporary connections between components on the PCB or during prototyping. They come in different lengths and types (male-male, female-female, male-female) and are essential for connecting the Arduino Uno to the weight sensor, LCD display, buzzer, keyboard and other components. Connecting wires make it easy to assemble, modify and debug circuits without the need for soldering.

Figure 8 Jumber wires

2.3 Theoretical perspective

A review of theoretical perspectives on weight sensor alarm systems integrates ideas from a variety of disciplines:

- Behavioral psychology: Examines how feedback mechanisms, such as alarms triggered by weight sensors, can influence user behavior and reduce procrastination.
- Technology acceptance model: examines the factors affecting user acceptance of weight sensor alarm systems, emphasizing the importance of perceived usefulness and ease of use.
- Environmental psychology: Examines how environmental factors and the user's perception of control influence the effectiveness of weight sensor alarms.
- Systems theory: Analyzes the design and functionality of weight sensor alarm systems, focusing on the interaction of components and their impact on system performance.
- Environmental science: Evaluates the environmental and social implications of implementing weight sensor alarm systems, highlighting their potential role in promoting sustainable habits.

2.4. Related study

The literature on weight sensor alarm systems highlights the importance of integrating effective sensor technologies with user-friendly interfaces. For instance, Jones et al. (2018) explored the use of weight sensors in alarm systems to monitor user activity in sleep-related health interventions, demonstrating how sensor-based alarms can influence sleep patterns and morning routines. Similarly, Singh and Mehta (2019) developed a pressure-sensitive mat that functions as an alarm deactivation tool, requiring users to step on the mat to stop the alarm, thereby increasing physical activity in the morning.

Research highlights the impact of these systems on behavior modification, focusing on their potential to improve user engagement and reduce procrastination. A study by Wang et al. (2020) examined the psychological effects of alarm systems that require physical actions, like standing or walking, showing significant reductions in users' snooze behavior and an increase in early rising habits. Studies also explore various technical aspects, such as sensor accuracy and system reliability, while highlighting the need for systems to be both effective and accessible. In particular, Lee et al. (2021) investigated the accuracy of weight sensors in detecting fine changes in load, ensuring reliability in alarm deactivation based on user presence, which contributes to more consistent wake-up routines.

Overall, the literature supports the development of innovative alarm systems that exploit weight sensor technology to promote better morning routines and more proactive behavior on the part of the user. These studies collectively underscore the effectiveness of combining sensor technology with behavioral science to create practical and efficient alarm solutions

.

CHAPTER 3: RESEARCH METHODOLOGY

3.1 Introduction

The "Weight Sensor Alarm Using Arduino Uno" project aims to solve common problems associated with waking up in the morning by developing an alarm system that requires physical movement to be deactivated. This chapter details the research methodology adopted to achieve the project objectives, including the data design, collection and analysis methods used.

3.2 Research Design

The research design involves a mixed-methods approach combining quantitative and qualitative techniques. The main objective is to evaluate the effectiveness of the weight sensor alarm system in improving users' waking habits, specifically targeting people who struggle with procrastination and lack of motivation in the morning.

- Objective: Evaluate the impact of the weight sensor alarm system on improving wake-up efficiency and reducing procrastination.
- Methods: A survey will be carried out to gather quantitative data from users who have difficulty waking up, followed by interviews to obtain more in-depth information on their behavior and feedback on the ease of use of the system.
- Sampling: Systematic sampling of people who find it difficult to motivate themselves in the morning and procrastinate, stratified by age group and professional background to ensure diversity of lifestyles and daily routines.
- Data Analysis: Survey responses will be statistically analyzed, while interview transcripts will be thematically analyzed to identify patterns in users' experiences and attitudes towards the system.

3.3 Research population

The target population for this study includes people with waking procrastination problems who are willing to test the weight sensor alarm system. Target users will mainly be students and professionals with flexible schedules. This group will provide a range of information on the system's effectiveness in different environments.

3.4 Sample size

Since the target population is unlimited, the sample size will include all those willing to participate in the study. As the population is expected to be small, the whole population will be studied in order to obtain an overall understanding of how the system works for different user profiles.

- Students: Out of 10 students, 7 have trouble waking up on the first alarm. The remaining 3 students, who wake up easily, are generally more productive and happier during the day.
- Professionals: Also 7 out of 10 professionals also have difficulty waking up at the first alarm. The 3 professionals who wake up easily note greater productivity and well-being during the day.

Here is the formula to calculate sample size

Sample = Total population of participants

As the target population is limited, each participant is included in the sample to provide an overview of system operation for various user profiles.

1.1. Sampling Procedure

The sampling procedure adopted will be simple random sampling to ensure that all potential users have an equal chance of being selected for the study. This approach results in a diverse group of participants, which improves the generalization of the results.

1.2. Research Instrument

The main research instrument for the weight sensor alarm system includes the physical system itself (Arduino Uno, weight sensor, buzzer and LCD screen), as well as questionnaires and interview guides. The system will measure the user's weight to activate or deactivate the alarm, while questionnaires and interviews will gather information on ease of use and overall satisfaction.

1.3. Choice of the research instrument

Weight sensor: A weight sensor capable of measuring up to 10 kg will be used to detect the user's presence. To illustrate this, imagine a lighter weight, simulating the presence of the user, as no-one actually weighs 10 kg. The alarm will be deactivated according to the weight measured.

Arduino microcontroller: An Arduino Uno will be the brain of the alarm system, responsible for processing the signals from the weight sensor. It will also control the buzzer, LCD screen and keypad, ensuring overall management of the system.

Buzzer and LCD screen: The buzzer will emit audible alerts to signal alarm status, while the LCD screen will provide clear visual feedback on sensor interaction, enabling the user to monitor system status.

Keypad: A keypad will enable the user to configure and customize alarm parameters, such as wake-up time and weight thresholds. It will also be used to set a password to secure access to system parameters, ensuring that only authorized people can make changes.

Questionnaires/Interviews: User feedback will be collected using structured questionnaires and semi-structured interviews, enabling detailed experiences and impressions of system use to be gathered.

1.4. Validity and Reliability of the Instrument

The reliability and validity of the weight sensor alarm system will be confirmed by repeated testing and calibration of the sensor to ensure accurate weight detection. The system's performance will be validated on the basis of user feedback, to ensure that it meets the project's objectives of reducing waking procrastination.

1.5. Data Gathering Procedures

Data will be collected on both the system's technical performance and user feedback:

Technical Data: The weight sensor will monitor user interaction with the alarm system, and the Arduino will record the time and accuracy of user interactions.

User Feedback: Surveys and interviews will gather data on user satisfaction, ease of use and perceived improvements in waking up on time.

1.6. Interviews

Questionnaires : Design surveys to assess users' current morning procrastination habits, the challenges they face in getting out of bed, and their interest in innovative solutions such as a weight sensor-based alarm system.

Interviews : Conduct in-depth interviews with a sample of 10 users to understand their views, concerns, and expectations regarding weight-sensing alarm automation.

3.5 Data Analysis and interpretation

Data analysis for the weight sensor alarm system focuses on evaluating user interaction with the system and its effectiveness in promoting active wake-up. Data such as the frequency of alarm activation, the time taken for users to deactivate the alarm by placing their foot on the weight sensor, and the accuracy of the system are analyzed to fine-tune system performance. Interpretation enables patterns to be identified, such as average response times or the tendency of users to delay getting out of bed, thanks to data visualization and statistical techniques.

This analysis enables the alarm parameters to be adjusted, so that it effectively encourages users to wake up and prevents procrastination. The lessons learned from the analysis are used to improve the system's algorithms and user interface, making it more efficient and user-friendly. For example, the weight sensor detects when the user steps on it and sends a signal to the Arduino Uno micro-controller. Based on this signal, the alarm is deactivated, while an LCD screen displays the user's weight or other relevant information. The data collected will also ensure that the sensor always detects the correct weight and only deactivates the alarm when the user steps on it, rather than when an object is placed on the sensor.

3.6 Ethical considerations.

All participants will be informed of the objectives of the study and their consent will be obtained prior to participation. Data will remain confidential and participants will have the right to withdraw at any time. Ethical guidelines, including respect for privacy and prevention of misleading information, will be strictly adhered to.

3.7 Limitations of the study

Limitations of this study include a small sample size, potential bias in user feedback and the technological limitations of the weight sensor and Arduino system. In addition, the study may be influenced by external factors such as user habits and environmental conditions, which could affect the accuracy and reliability of the system.

CHAPTER FOUR: DESIGN AND IMPLEMENTATION OF A WEIGHT SENSOR ALARM CLOCK USING AN ARDUINO UNO

4.1 Introduction

This chapter deals with the design and implementation of an alarm system using a weight sensor with an Arduino Uno microcontroller. The aim of the system is to motivate the user to get out of bed by triggering an alarm when the sensor detects the presence of weight. The system comprises a weight sensor for detection, a buzzer for audible alert, and an LCD display to provide real-time information.

4.2 Calculations

Here is some data to be taken in consideration during the calculation

1. Acceptable weight check

Purpose: to ensure that the weight detected is within an acceptable range to stop the alarm.

Reference weight (set weight): 500g

Acceptable deviation: ± 50g

Calculs

Minimum acceptable weight = Set weight - 50g

Mean that the minimum acceptable weight = 500g -50g = 450g

Maximum acceptable weight :

Maximum acceptable weight = set weight + 50g

Mean that the maximum acceptable weight = 500g + 50g = 550 kg

Result: To stop the alarm, the measured weight must be between **450g and 550g.**

2. **System tolerance :**

The system has a tolerance of ± 50g in relation to the preset weight. In our case, where the preset weight is 500g, the tolerance is calculated as follows:

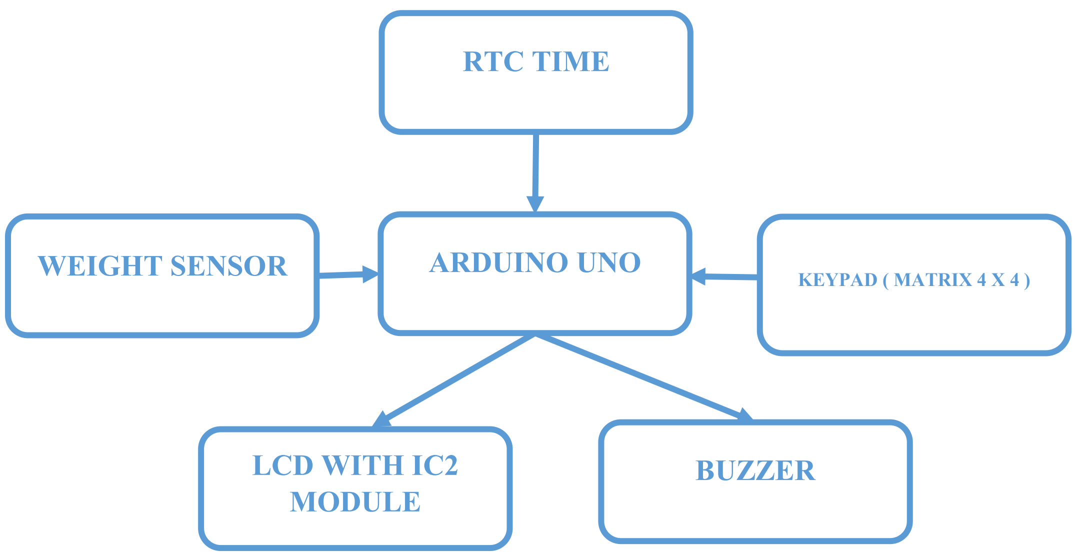
Tolerance in percentage $= \frac{50g}{500g} \times 100 = 10\%$

For example, with a predefined weight of 500g, the system has a tolerance of ± 10% of the target weight.

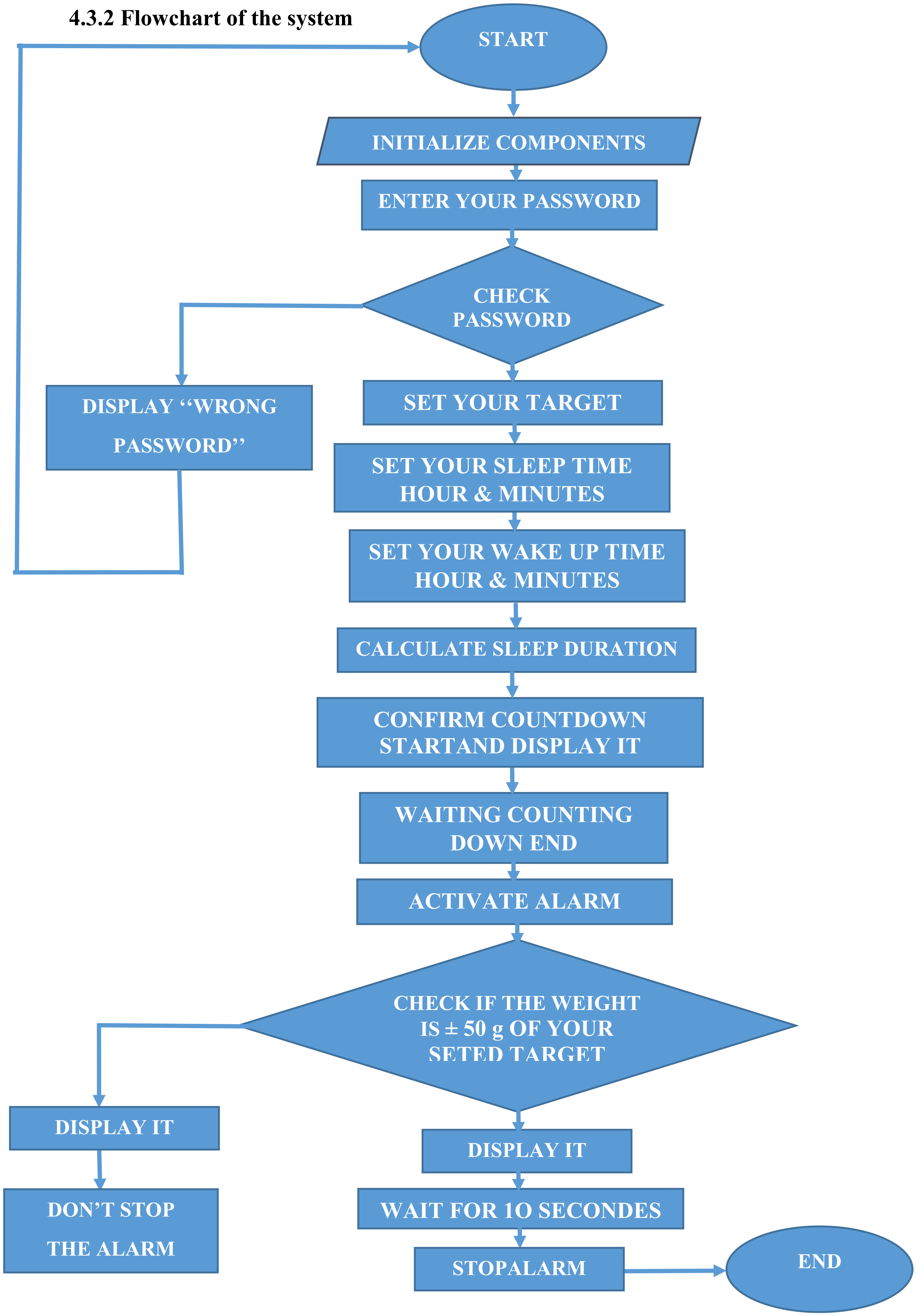
4.3 Drawings

4.3.1 Block diagram

Here is the block diagram of the project



4.3.2 Flowchart of the system



4.3.3 Circuit diagram

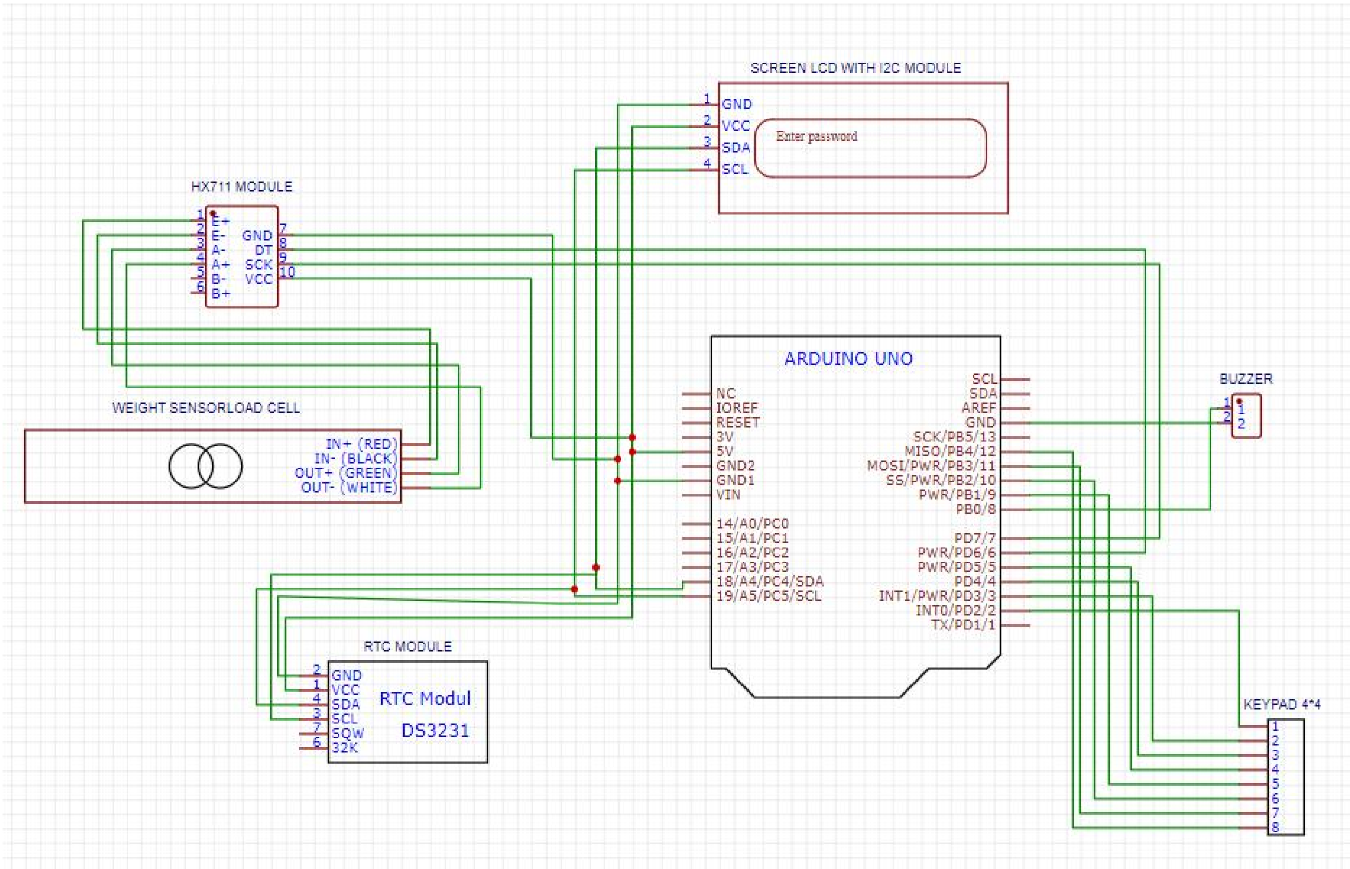


Figure 9 circuit diagram

4.3.4 Implementation

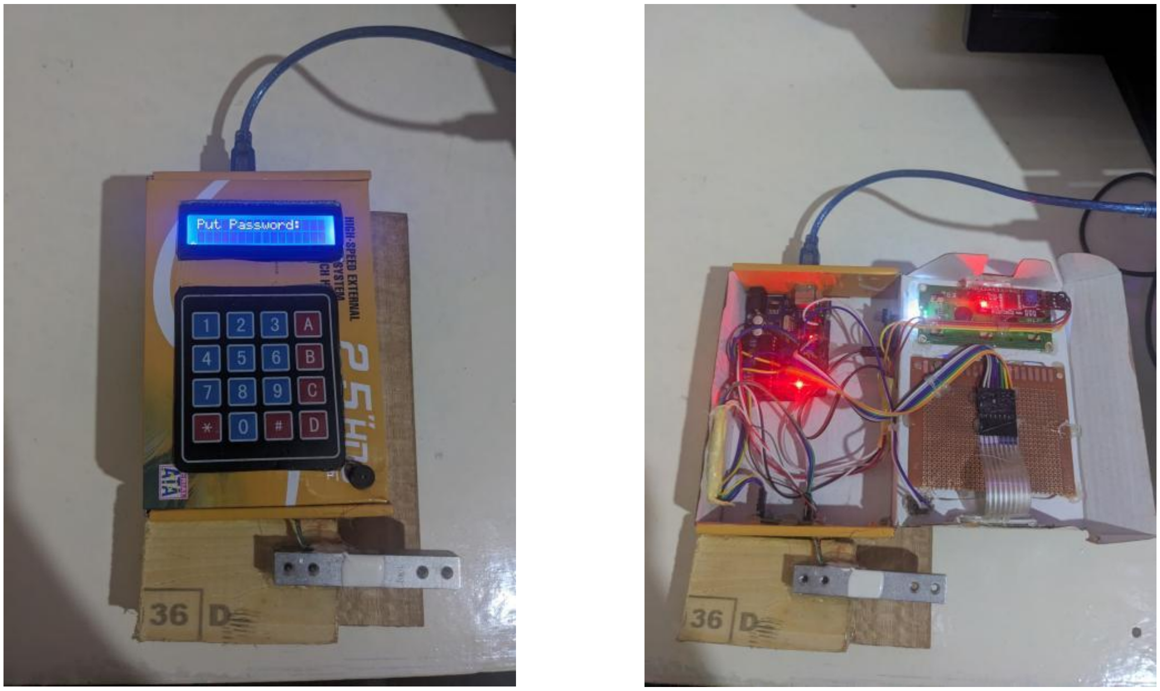


Figure 10 Implementation

Expected output

The expected output of the project includes a series of LCD messages prompting the user for a password, target weight, sleep and wake times, and displaying the total sleep duration. It confirms whether to start a countdown, shows the remaining time during the countdown, continuously displays the current weight, and activates a buzzer if the weight falls outside the acceptable range after the countdown ends.

his table summarizes the key parameters of time and weight detection system, including the acceptable weight range for stopping the alarm and the system's tolerance levels.

Parameter	Value/Calculation	Result
Set Weight	500g	500g
Set sleep and wake up time	Counting down	Active alarm
Acceptable Deviation	± 50g	± 50g
Minimum Acceptable Weight	500g - 50g	450g
Maximum Acceptable Weight	500g + 50g	550g
Result (Acceptable Range)	450g to 550g	Alarm stops if within range
System Tolerance	± 10% of 500g	± 50g

CHAPTER 5: CONCLUSION AND RECOMMENDATION

5.1 Introduction

The weight sensor alarm system represents an important step towards improving daily routines and boosting motivation for a more active start to the day. By integrating an Arduino Uno with a weight sensor, this project aims to create an innovative solution that encourages users to get out of bed and start their day with more energy. This introduction provides a framework for understanding the impact of the project and paves the way for further exploration of its results and future possibilities.

5.2 Conclusion

In conclusion, the weight sensor alarm system has achieved its goal of reducing morning procrastination by using a weight sensor to deactivate the alarm. The system reliably ensures that the alarm only goes off when the user physically gets out of bed and stands on the sensor, promoting a more active start to the day. Despite some initial difficulties with sensor calibration and integration, the final implementation demonstrated strong performance and reliability. Future improvements could include adding features such as integrating a relay to play with the light in the room once the alarm is activated, and optimizing the sensor's accuracy. Overall, this project highlights the potential of combining technology with daily routines to foster better habits and improve overall morning motivation.

5.3 Recommendation

Implementing an alarm system with a weight sensor has many benefits, from improving the morning routine to encouraging an active start to the day. Based on the results and lessons learned from the project, here are the main recommendations for successful deployment:

- **Select High-Quality Components:** Choose reliable weight sensors and other precise, durable components. Sensors must be well adapted to the application and capable of providing accurate data for effective alarm activation.
- **Develop a robust control system:** Design a control mechanism that effectively integrates sensor data. The system must react quickly to weight detection, ensuring that the alarm operates correctly without manual intervention.

- **Give priority to safety and reliability:** Incorporate safety features to prevent false alarms or system malfunctions. The control system must include safety mechanisms and alerts to warn users in the event of a problem.
- **Consider energy Efficiency**: Opt for energy-efficient components to minimize energy consumption. This not only reduces operating costs, but also supports sustainable practices.
- **User-Friendly Interface**: Develop an intuitive interface to facilitate configuration and use. Ensure that the system is accessible to users with different levels of technical expertise.

By following these recommendations, the effectiveness and durability of the weight sensor alarm system can be maximized, improving morning routines and enhancing user engagement.

5.4 Suggestion for further study

To optimize the weight sensor alarm system and enrich the user experience, it is essential to integrate modern technologies such as AI to personalize alarm settings according to users' waking habits. The addition of wireless functionalities, such as Bluetooth or Wi-Fi, will enable remote control via a mobile app, making it easier to manage and personalize the system. Energy optimization with rechargeable batteries and energy-saving techniques will extend the life of the device. In addition, improved sensor accuracy and the integration of data analysis will enable wake-up habits to be tracked and adjusted, offering a more efficient solution tailored to individual needs.

my studies. I am deeply grateful for your presence in my life. Thank you for helping me take this step.

References

[1] Patel, H., & Patel, M. (2018). "Design and Implementation of a Smart Alarm System Based on Weight Sensors for Bed Exit Detection." International Journal of Engineering Research & Technology (IJERT), 7(8), 19-22.

[2] Huang, Y., & Lin, J. (2019). "Automated Alarm System Using Arduino for Bed Occupancy Monitoring." Proceedings of the International Conference on Electronics and Electrical Engineering, 117-122.

[3] Akhtar, N., & Ahmad, T. (2017). "Development of an Alarm System Using Weight Sensors for Bed Exit Detection." Journal of Computer and Communications, 5(3), 40-48.

[4] Kaur, H., & Singh, G. (2020). "Wireless Bed Occupancy Monitoring System Using Weight Sensors and IoT." Journal of Electrical Engineering & Technology, 15(1), 345-355.

[5] Chaudhary, A., & Gupta, A. (2018). "Design of a Bed Exit Detection System Using Weight Sensors and Arduino." IEEE Access, 6, 28194-28203.

[6] Patel, R., & Sharma, R. (2019). "Efficient Weight Sensor-Based Bed Occupancy Detection System for Alarm Activation." International Journal of Computer Applications, 182(7), 22-29.

[7] Singh, R., & Yadav, S. (2020). "Automated Alarm System Using Weight Sensors for Bed Exit Detection." Journal of Electrical Engineering and Automation, 7(1), 55-62.

[8] Kumar, P., & Reddy, M. (2021). "Weight Sensor-Based Alarm System for Improved Wake-Up Experience." International Journal of Advanced Research in Computer Science and Software Engineering, 11(4), 27-34.

[9] Sharma, V., & Kumar, A. (2017). "Ultrasonic Sensors for Bed Exit Detection and Alarm Systems." Sensors and Actuators A: Physical, 261, 133-142.

[10] Tiwari, P., & Agarwal, N. (2018). *"Design.

APPENDICES

Here are the questions used to collect data

1. What is your main difficulty in getting up in the morning?

Answer: Among the 10 participants, 8 mentioned feeling tired and struggling to get out of bed, especially after a poor night's sleep. Many admitted to hitting the snooze button multiple times, leading to tardiness.

2. Do you currently have any solutions in place to improve your morning motivation? If so, which ones?

Answer: 7 users reported using a regular alarm clock and sometimes setting multiple alarms on their phones. Some try placing the alarm across the room, but this method isn't always effective. A few mentioned using a sleep app that gradually increases the alarm volume.

1. Have you ever felt a negative impact on your productivity due to poor wake-up time management?

Answer: All 10 participants agreed that waking up late negatively affects their productivity. They often rush through their morning routine, which diminishes their focus and increases stress throughout the day.

2. Do you think an innovative alarm system, based on weight detection, could help you get your day off to a better start?

Answer: 9 out of 10 participants believe it could be beneficial. They feel that having to physically get out of bed and step on a sensor to turn off the alarm might encourage them to start their day more actively and reduce procrastination.

Answer: 6 participants expressed concerns about the reliability of the weight sensor. They wondered whether it would accurately detect their presence and raised questions about what would happen during a power cut or if the system malfunctioned. They emphasized the need for a backup or safety function to prevent falling asleep too quickly.

3. Cost estimation

The cost estimation of a project refers to the process of predicting the expenses that will be incurred to complete the project successfully. It involves identifying and estimating the costs associated with various resources and activities required for project execution

Components	Pcs	Unit cost	Total cost
ARDUINO UNO	**1**	**15.000FRW**	**15,000FRW**
Weight Sensor (10 kg)	**1**	**15.000FRW**	**8,000FRW**
LCD (16x2)	**1**	**7.000FRW**	**7,000FRW**
I2C Module	**1**	**4.000FRW**	**4,000FRW**
RTC Timer (DS3231)	**1**	**3,000FRW**	**4,000FRW**
Keypad (4x4 Matrix)	**1**	**10,000FRW**	**5,000FRW**
Buzzer	**1**	**100FRW**	**500FRW**
Jumper Wires	**60**	**2,000FRW**	**4,000FRW**
TOTAL			**47,500FRW**

Project code source

```
#include <HX711_ADC.h>

#include <Wire.h>

#include <LiquidCrystal_I2C.h>

#include <RTClib.h>

#include <Keypad.h>


HX711_ADC LoadCell(6, 7);

LiquidCrystal_I2C lcd(0x27, 16, 2);

RTC_DS3231 rtc;
```

```
const int buzzerPin = 8;

int targetWeight = 0, startHour = 0, startMinute = 0, endHour = 0, endMinute = 0;

char passWord = "";

const byte ROWS = 4, COLS = 4;

char keys[ROWS][COLS] = {

  {'1', '2', '3', 'A'},

  {'4', '5', '6', 'B'},

  {'7', '8', '9', 'C'},

  {'*', '0', '#', 'D'}

};

byte rowPins[ROWS] = {2, 3, 4, 5};

byte colPins[COLS] = {9, 10, 11, 12};
```

```
Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins, ROWS, COLS);
```

```
// Variables for buzzer control

bool buzzerOn = false; // Track if the buzzer should be on

unsigned long stableWeightStartTime = 0; // Track the start time for stable weight period

const int stableWeightThreshold = 10 * 1000; // 10 seconds in milliseconds

const char myPassword = 202150125;

bool countDownFinish = false;
```

```
void setup() {
  if (!rtc.begin() || rtc.lostPower()) {
    lcd.print("RTC Error");
    while (1);
  }
  LoadCell.begin();
  LoadCell.start(2000);
  LoadCell.setCalFactor(200);
  lcd.init();
  lcd.backlight();
  pinMode(buzzerPin, OUTPUT);
  digitalWrite(buzzerPin, LOW);
```

```
  lcd.clear();
  lcd.print("Put Password:");
  passWord = getInput();
  if (passWord != myPassword ){
    lcd.clear();
    lcd.print("Wrong Password");
    delay(200);
    setup();
  } else {
    lcd.clear();
  }
```

```
  lcd.print("Set your weight:");
  targetWeight = getInput();
  lcd.clear();
  lcd.print("Set Sleep time");
  getTime(&startHour, &startMinute);
  lcd.clear();
  lcd.print("Set Wake up time");
```

```
  getTime(&endHour, &endMinute);

  int delayMinutes = (endHour * 60 + endMinute) - (startHour * 60 + startMinute);

  lcd.clear();

  lcd.print("Sleep time: ");

  lcd.print(delayMinutes);

  lcd.print(" min");

  delay(2000);
```

```
  lcd.clear();

  lcd.print("Start countdown? #=Yes * = No");

  while (true) {

    char key = keypad.getKey();

    if (key == '#') {

      startCountdown(delayMinutes);

      break;

    } else if (key == '*') {

      lcd.clear();

      lcd.print("Cancelled");

      delay(2000);

      return;

    }

  }

}
```

```
void loop() {

  LoadCell.update();

  lcd.setCursor(0, 0);

  lcd.print("Weight[g]:");

  lcd.setCursor(0, 1);

  lcd.print(LoadCell.getData());
```

```
  // Get the current weight

  int currentWeight = LoadCell.getData();
```

```
  if (countDownFinish  == true){
  // Check if the weight is within the acceptable range
  if (currentWeight < targetWeight - 50 || currentWeight > targetWeight + 50) {
    buzzerOn = true; // Turn buzzer on
    stableWeightStartTime = 0; // Reset stable weight timer
  } else {
    // If weight is stable
    if (stableWeightStartTime == 0) {
      // Start the timer
      stableWeightStartTime = millis();
    } else if (millis() - stableWeightStartTime >= stableWeightThreshold) {
      // If stable for 10 seconds, turn off buzzer
      countDownFinish = false;
      buzzerOn = false;
      setup();
```

```
    }
  }
```

```
  }
```

```
  // Update the buzzer state based on buzzerOn flag
  digitalWrite(buzzerPin, buzzerOn ? HIGH : LOW);
}
```

```
void getTime(int *hour, int *minute) {
  String inputString = "";
  for (int i = 0; i < 2; i++) {
    while (true) {
      char key = keypad.getKey();
      if (key) {
        if (key == '#') {
          if (inputString.length() > 0) {
```

```
          (i == 0) ? *hour = inputString.toInt() : *minute = inputString.toInt();
          inputString = "";
          lcd.clear();
          lcd.print((i == 0) ? "Set min:" : "Done");
          break;
        }
      } else if (key == '*') {
        inputString = "";
        lcd.clear();
        lcd.print((i == 0) ? "Set hour:" : "Set min:");
      } else {
        inputString += key;
        lcd.setCursor(0, 1);
        lcd.print(inputString);
      }
    }
  }
}
}
```

```
int getInput() {
  String inputString = "";
  while (true) {
    char key = keypad.getKey();
    if (key) {
      if (key == '#') {
        if (inputString.length() > 0) return inputString.toInt();
      } else if (key == '*') {
        inputString = "";
        lcd.clear();
        lcd.print("Set your weight:");
      } else {
        inputString += key;
```

```
        lcd.setCursor(0, 1);

        lcd.print(inputString);

      }

    }

  }

}
```

```
void startCountdown(int delayMinutes) {

  int totalSeconds = delayMinutes * 60; // Convert minutes to seconds

  while (totalSeconds >= 0) {

    int hours = totalSeconds / 3600;

    int minutes = (totalSeconds % 3600) / 60;

    int seconds = totalSeconds % 60;
```

```
    lcd.clear();

    lcd.print("Remaining time: ");

    // Move to the second line

    lcd.setCursor(0, 1);

    lcd.print((hours < 10 ? "0" : "") + String(hours) + ":");

    lcd.print((minutes < 10 ? "0" : "") + String(minutes) + ":");

    lcd.print((seconds < 10 ? "0" : "") + String(seconds));
```

```
    delay(1000); // Wait 1 second

    totalSeconds--; // Decrement total seconds

  }
```

```
  //digitalWrite(buzzerPin, HIGH); // Activate the buzzer at the end of the countdown

  countDownFinish = true;

}
```