CMPE443: PRINCIPLES OF EMBEDDED SYSTEMS DESIGN
(FALL'23)

TRASH BIN 2.0 : AN INDOOR SMART GARBAGE HANDLING SYSTEM WITH ENHANCED FEATURES FOR HYGIENE, SAFETY AND CONVENIENCE

PROTOTYPE REPORT

Submitted by

GROUP DESIGN R

Necdet Can Uzduran (2019400195)

Melih Gezer (2020400156)

Arda Kabadayı (2019400129)

Süleyman Melih Portakal (2019400240)

Hakan Emre Aktaş (2020400351)

Batuhan Çetin (2019400117)

Computer Engineering Department

Boğaziçi University

January 15, 2024

Contents

Abstract

In this project proposal, we focused on addressing the widespread challenge of trash bins overflowing and tripping over. The problem we are addressing leads to the creation of unsightly and unhygienic environments, affecting our daily lives, public spaces, and households. In order to solve this problem, we propose a brand new product, a smart bin which is an embedded system with various specialized input and output components and software to efficiently and securely manage and monitor the functionality of trash bins. Our project offers a range of functionalities, including detecting whether the bin is tripped over, locking the lid in order to prevent spreading, detecting liquid leaks, touch-free lid opening, measuring weight and fullness, remote monitoring and control. By implementing these features, we aim to provide a comprehensive solution to the problems associated with traditional trash bins.

As embedded systems are typically application-specific, our proposed Smart Bin system is also designed to be that way, with input sensors, output devices and other components are selected specifically for our embedded system's functions. Although we will mention properties of our system in more detail later on, it is appropriate to give the reader some insights about our Smart Bin at this stage. Our Smart Bin operates in a timely manner, ensuring swift responses to various events, such as bin tripping and leakage from the trash bag. While designing our system conceptually, we have also considered cost-effectiveness, size constraints, power consumption, and reliability. Additionally, we offer connectivity for remote monitoring and control, in order to provide a more convenient management for the user.

The goals of our system consist of precise level monitoring, automated lid locking, touch-free lid opening, fall/trip detection using a gyroscope, leak detection, an alert/notification system featuring LED indicators, and remote notifications. This solution aims to improve convenience, hygiene, and security in various settings, including public places, personal spaces, and commercial areas. By addressing these challenges, we seek to enhance the quality of life and the environment, offering a more convenient and smarter solution relative to traditional trash bins.

In this prototype, we used 3 ultrasonic sensor. One for automated lid opening and locking, one for fall detection instead of gyroscope and one for measuring fullness of the bin. We displayed fullness of the bin with 7 segment displays as percent. We used one flame sensor and one gas sensor for increasing safety of the bin. We are

measuring weight of the trash bin with a weight sensor and prints it to the console in order inform officials. In order to warn scavengers in case of liquid leakage we used raindrop sensor and a led. Most sensors prints their status to the console in case of warning with LPUART. Lastly we used sound sensor to detect living creatures inside of the bin. With these features we implemented convenient, safe and automized trash bin prototype.

Changes

Table 1: Component Changes

Component in the Original Proposal	Component in the Prototype	Reason
–	Sound Sensor	We added sound sensor to detect there is a sound in the trash bin.
–	Flame Sensor	We proposed flame sensor in order to detect fire in the trash bin.
–	Gas Sensor	We added gas sensor to detect there is gas or smoke in the trash bin.
Gyro Sensor	Ultrasound Sensor	We changed gyro sensor to ultrasound sensor in order to detect fall of the bin but functionality stays the same.

Table 2: Minor Functionality Changes

Functionality in the Original Proposal	Functionality in the Prototype	Reason
–	Sound Detection	Sound sensor listens the sounds in the trash bin in order to detect a living creature like a cat or a baby.
–	Fire Detection	We added fire detection in order to increase safety of the trash bin.
–	Gas Detection	We added gas and smoke detection in order to increase safety of the trash bin.

. .

1. Overview

Filename: *chapters/Description.tex*

Describe the your implementation.

Draw the final block diagram.

Name the components of your implementation. Write the part number explicitly. Give a reference to the datasheet of each component [1].

Describe the implemented behavior.

If there are changes, put separate sections for each change.

Fill Table 1.1 to provide an overview of your implementation.

Table 1.1: Implementation Overview

Component	Peripheral	Pins	Access Method	global variables	Shared functions
Sound Sensor	ADC, GPIO, TIM	PC1	check_sound	sound_dig_value, index, alert sound	send_alert, ADC1_2_IRQHandler
Flame Sensor	ADC, GPIO, TIM	PC0	check_fire	flame_dig_value, index, is_fire, alert_fire	send_alert, ADC1_2_IRQHandler
Gas Sensor	ADC, GPIO,TIM	PC3	check_gas	gas_dig_value, index, is_gas, alert_gas	send_alert, ADC1_2_IRQHandler
Raindrop Sensor	GPIO	PC12	check_rain_drop	–	–
Weight Sensor	GPIO, TIM, LPUART	PC10, PC11	weight_read_data	–	wait_nano
Ultrasonic Sensors	GPIO, TIM, LPUART	PA0:3, PB3:4, PB14:15, PG0:1	TIM2,3,6,7,15,16 Handlers	–	–
Seven Segment Displays	GPIO	PD0:7	–	–	–
Servo motors	GPIO, TIM	PA6:7	TIM16_IRQHandler	–	–
Buzzer	GPIO	PC6	–	–	check_fire
LEDs	GPIO	PG0:1, PD9	–	–	TIM16_IRQHandler, check_rain_drop

1.1 Implementation

Our implementation contains various sensors and actuators to generate a smart bin system. Firstly, our smart bin has a touchless opening lid functionality. By using a ultrasonic sensor we can detect whether someone is trying to open the smart bin and the bin is opened via servo motor. Additionally, our bin has a weight sensor to measure the weight of the trash. It also includes an ultrasonic sensor to measure the height of the bin and this height is shown by using two 7 segment display. Both of these measurements indicates the fullness of the smart bin. And whenever a threshold is exceeded it warns the user. Our embedded system can detect liquid leaking from the trash bin via raindrop sensor. When our sensor detects the liquid, it will turn on the blue light. Also we put some sensors to prevent huge disasters that can be created due to trash or some external factors. The first one is flame sensor. Whenever the trash burns our flame sensor immediately detects the flame and activates the buzzer. The second one is gas sensor. If the trash remains in the bin for a long time it can emit unpleasant gases such as methane gas. Whenever the trash starts to generate harmful gases excessively, our sensor detects it and again activates the buzzer. So we think these cases as emergency cases and it requires immediate actions. Finally our smart bin has a sound sensor. This sensor is used detecting a live creature in the bin such as a kitten or a baby. In this case we send a message to the user via UART to indicate that a sound is coming from inside the bin. For itemized representation you can see the below part.

1. Touchless Opening Lid:

    - With the ultrasonic sensor on top of the trash bin, the lid of the bin can be opened with the servo if someone approaches with their hand.

2. Weight Measurement:

    - Via weight sensor, we measure the weight of the trash.
    - We put a weight threshold and after exceeding it we inform the user about the fullness.

3. Height Measurement:

    - We used an ultrasonic sensor to measure the height of the trash inside the bin.
    - The measured height is displayed on two 7 segment displays to provide a information about the bin fullness.
    - We put a height threshold and after exceeding it we inform the user about the fullness.

4. Flame Detection

    - Used to detect the presence of flame inside the smart bin.
    - If a flame is detected, we activate the buzzer

5. Gas Detection

    - Used to detect the presence of gas inside the smart bin.

- If a smoke is detected, we activate the buzzer

6. Sound Detection

- Used to detect the presence of sound inside the smart bin.
- If a sound is detected, user is informed via UART.

7. Rain Drop Detection

- Used to detect the presence of a liquid bottom of the smart bin.
- If a liquid is detected, blue led is activated.

8. Fall Detection

- Used to detect the whether the bin has fallen or not
- If the trash bin falls a buzzer is activated.

1.2 Final Block Diagram

1.3 Components

- **3x Ultrasonic Sensor :** used for trash level detection, opening the lid automatically and detecting falling of the bin.
  - Sensor: HC-SR04 Ultrasonic Distance Sensor
  - Datasheet: https://datasheetspdf.com/pdf-file/1380136/ETC/HC-SR04/1

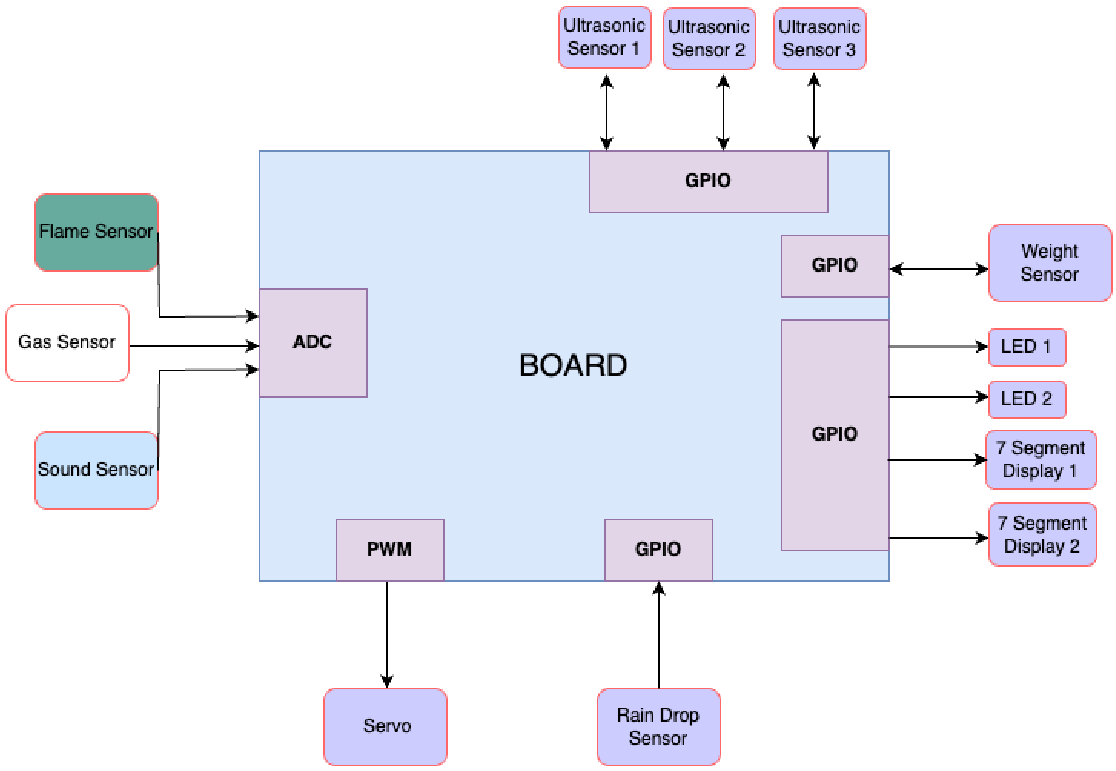


Figure 1.1: Final Block Diagram

Figure 1.2: Ultrasonic Sensor

- **Raindrop Sensor :**
  - Sensor: MH-RD Raindrop Sensor Module
  - Datasheet: https://components101.com/sensors/rain-drop-sensor-module

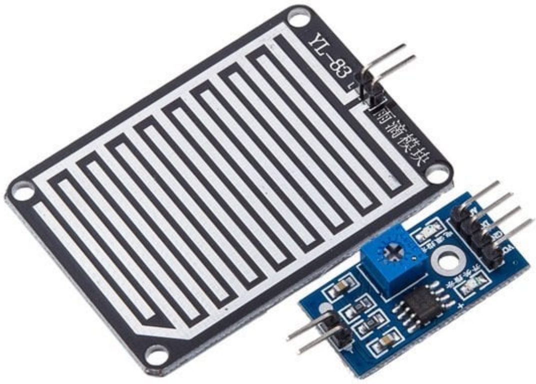


Figure 1.3: Raindrop Sensor

- **Weight Sensor :**
  - Sensor: Hx711 Module Weight Kit
  - Datasheet: https://www.digikey.com/htmldatasheets/production/1836471/0/0/1/hx711.html

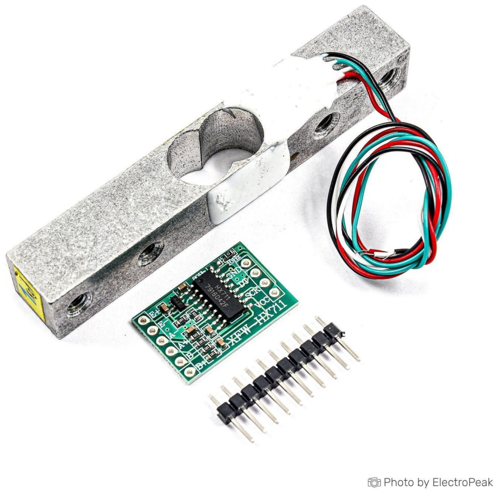


Figure 1.4: Weight Sensor

- **Sound Sensor :**
  - Sensor: KY-038 Sound Sensor Module
  - Datasheet: https://datasheetspdf.com/pdf-file/1402048/Joy-IT/KY-038/1

Figure 1.5: Sound Sensor

- **Gas Sensor :**
  - Sensor: MQ-9 Gas Sensor Module
  - Datasheet: https://www.haoyuelectronics.com/Attachment/MQ-9/MQ9.pdf

Figure 1.6: Gas Sensor

- **Flame Sensor :**
  - Sensor: SEN16 Flame Sensor Module
  - Datasheet: https://www.rajguruelectronics.com/Product/13200/A132002_Flame%20sensor

Figure 1.7: Flame Sensor

- **Computer/Bluetooth :** used to lock or unlock the lid manually and turn off the buzzer.

- **Buzzer :** used to warn in case of fall, gas and fire.

Figure 1.8: Buzzer

- **2x LED :** used to warn in case of humidity and opening of lid.

Figure 1.9: LED

- **2x Seven Segment Display :** used to display fullness of the trash bin.

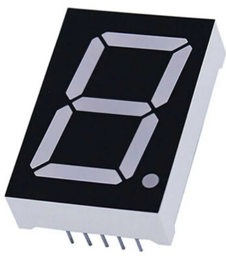


Figure 1.10: Seven Segment Display

1.4 Implemented Behaviour

We implemented many functionalities and warning mechanisms for the trash bin. Firstly there is an ultrasound sensor in front of the lid. So when a person brings his/her hands closer to the lid in order to throw a trash, sensor sees the hand and opens the lid automatically and turn on green light. If sensor doesn't see anything in 3 seconds interval, it means that throwing trash to the bin process is finished and it closes the lid automatically and converts green light to red light. The trash bin has a weight sensor. under of it and it measures the weight of the trashes and prints it to the console through LPUART. There is a ultrasound sensor inside of the lid to measure fullness of the trash bin and display this fullness percent through two seven segment display. Also, there is another ultrasound sensor under the bin to detect trash bin is in the correct vertical position or it fell to the floor. If the trash bin falls, it enables the buzzer. There are four sensors inside of the trash bin. First one is a flame sensor that detects there is fire inside of the bin.

If there is fire, it enables the buzzer to warn the environment, also prints to the console through LPUART. Second one is a sound sensor that detects there is a sound inside of the bin. If there is a sound, it can mean that there is a living thing like a cat in the bin and prints a warning to the console. Third one is gas detector. It detects smoke or gas like carbonmonoxide and enables the buzzer, prints to the console if there is gas. Last sensor is raindrop sensor. It detects the humidity and turns on blue light if there is wet in the trash bin to warn.

1.5 Changes

We preserved functionalities and components of the project mostly but there are some little additions. In order to detect fire, smoke and gas in the trash bin and increase safety of the project, we added two sensors: flame sensor and gas sensor. Both of these sensors work as ADC. According to output of these sensors the trash bin gives a warning with buzzer. Also in order to detect there is a living thing inside the trash bin like a cat, we added sound sensor inside of the trash bin. The trash bin prints to the UART console according to the level of the sound in the trash bin.

2. Detailed Explanation

2.1 Schematic Diagram

2.2 Sequence Diagram

2.3 Pseudo Code

Algorithm 1 `Main Algorithm`

$initialize()$
$turnOnRed()$
$enableInterrupts()$
$i = 0$
while $systemOn()$ **do**
  if $which7s$ **then**
    $writeToFirst7s()$
  else
    $writeToSecond7s()$
  end if
  $checkAdc()$
  if $isFall\ or\ isFire\ or\ isGas$ **then**
    $activateBuzzer()$
  else
    $deactivateBuzzer()$
  end if
  if $weightReadBit()! = 1$ **then**
    $weightReadData()$
  end if
  if $checkRainDrop() == 0$ **then**
    $turnOnBlue()$
  else
    $turnOffBlue()$
  end if
end while

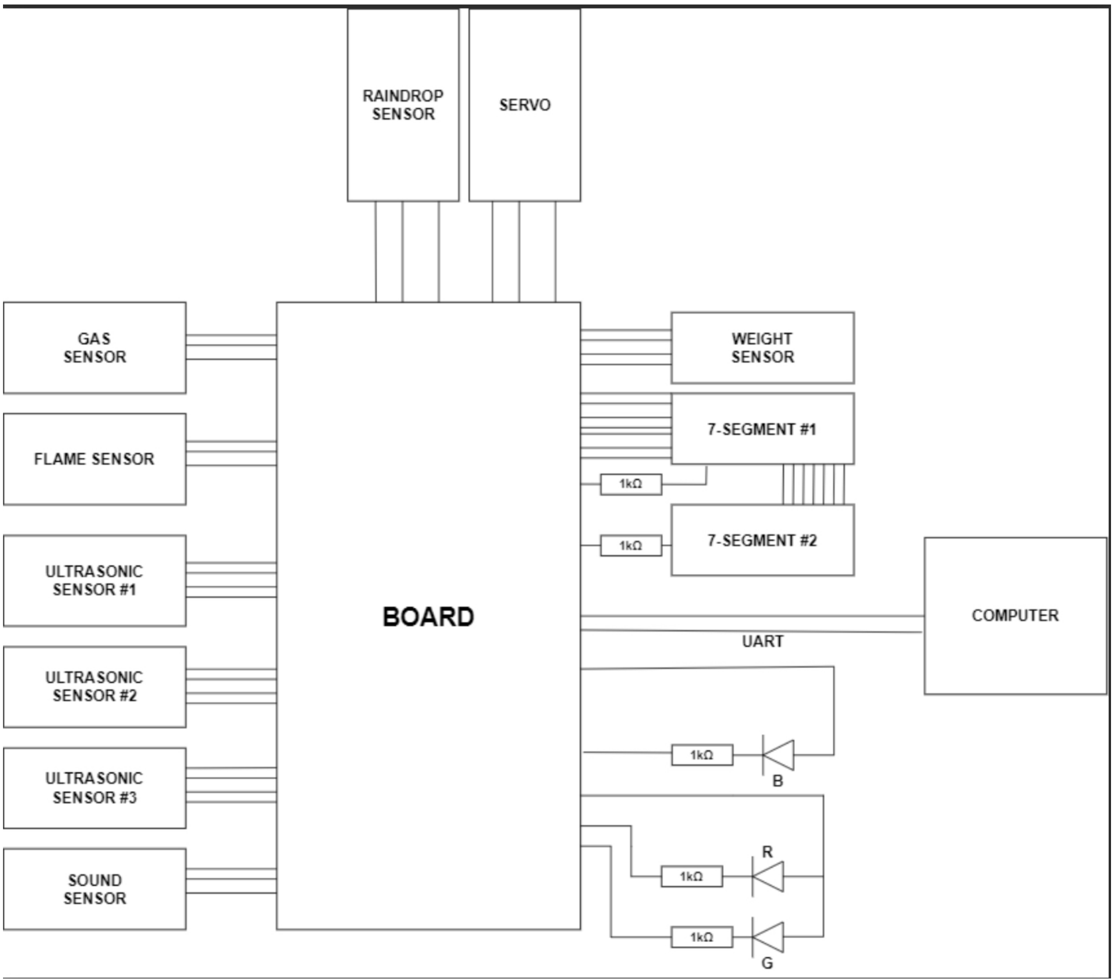


Figure 2.1: Schematic Diagram

Algorithm 2 `Fire Detection`

if $flameDigValue() == 0$ **then**
  $isFire = 1$
  $sendAlert(f)$
else
  $isFire = 0$
end if

Algorithm 3 `Gas Detection`

if $gasDigValue() > 1000$ **then**
  $isGas = 1$
  $sendAlert(g)$
else
  $isGas = 0$
end if

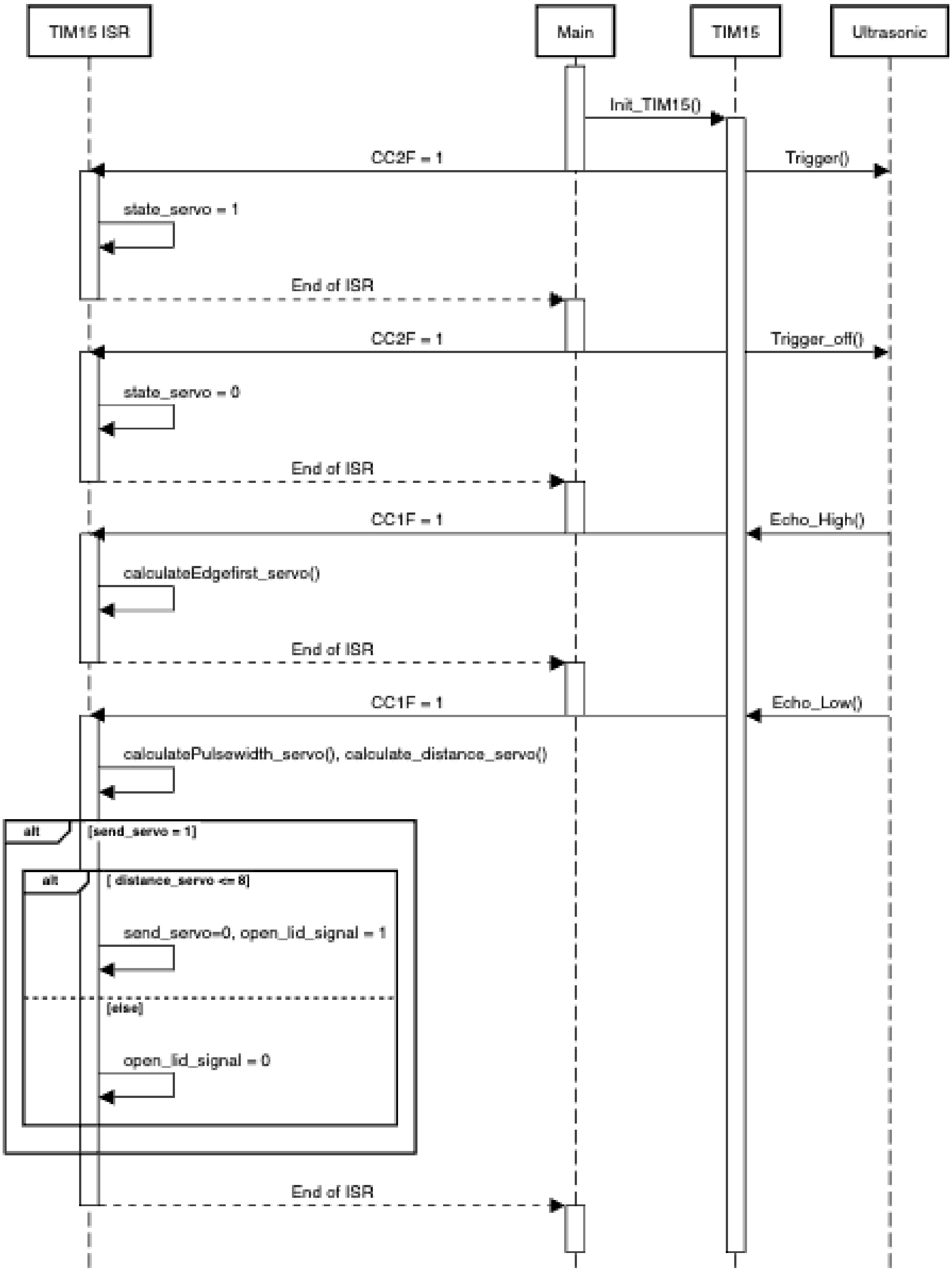


Figure 2.2: Sequence Diagram for Opening Lid Ultrasonic Part

Algorithm 4 `Sound Detection`

if $soundDigValue() < 3800$ **then**
  $isSound = 1$
  $sendAlert(s)$
else
  $isSound = 0$
end if

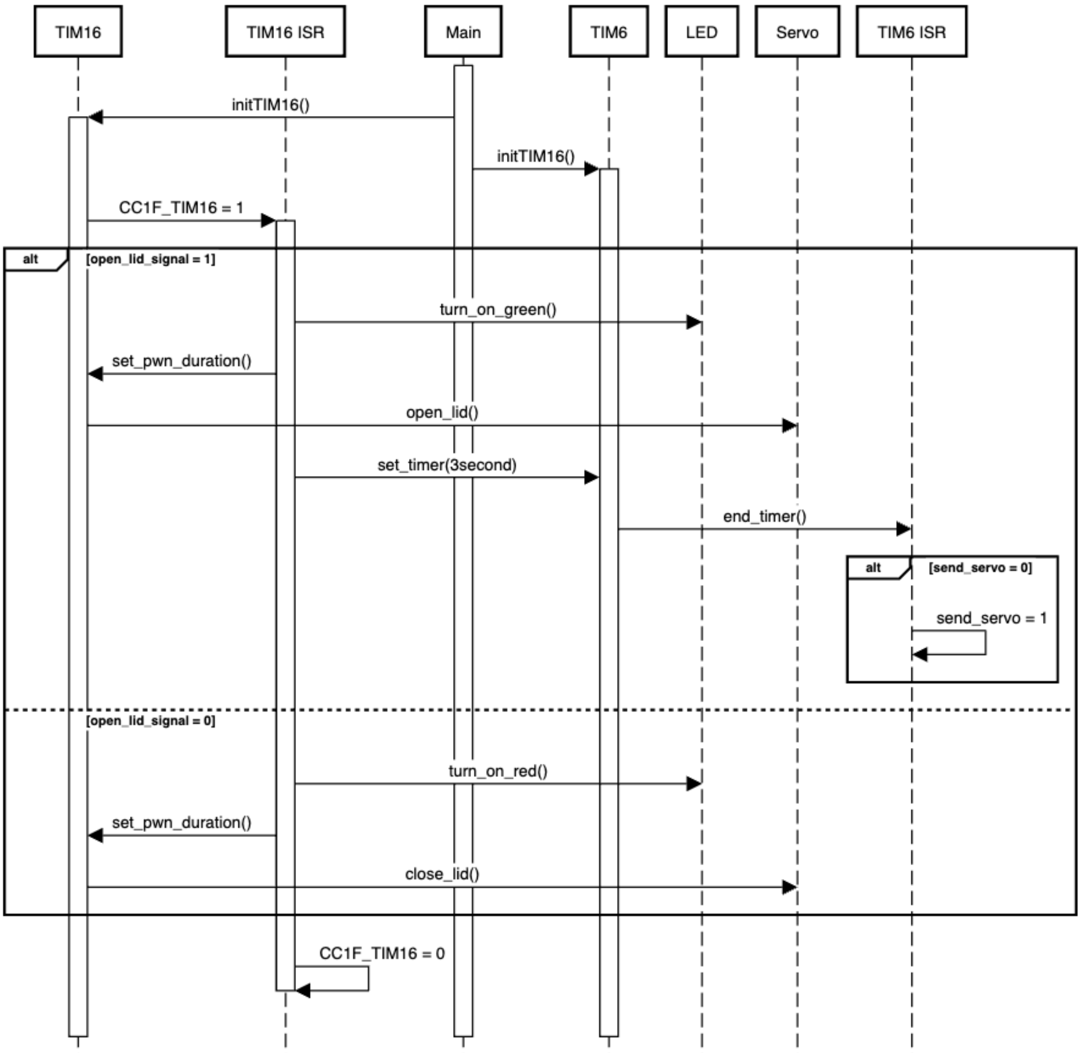


Figure 2.3: Sequence Diagram for Opening Lid Servo Part

Algorithm 5 `Weight Measurement`

$data = 0$
for $i \leftarrow 23$ **to** 0 **do**
    $giveHigh()$
    $waitNano()$
    $readWeightBit()$
    $giveLow()$
    $waitNano()$
end for
$giveHigh()$
$waitNano()$
$giveLow()$
$waitNano()$
$extendDataFor2sComplement(data)$
$weightData = data * 0.002140 - 958.7$

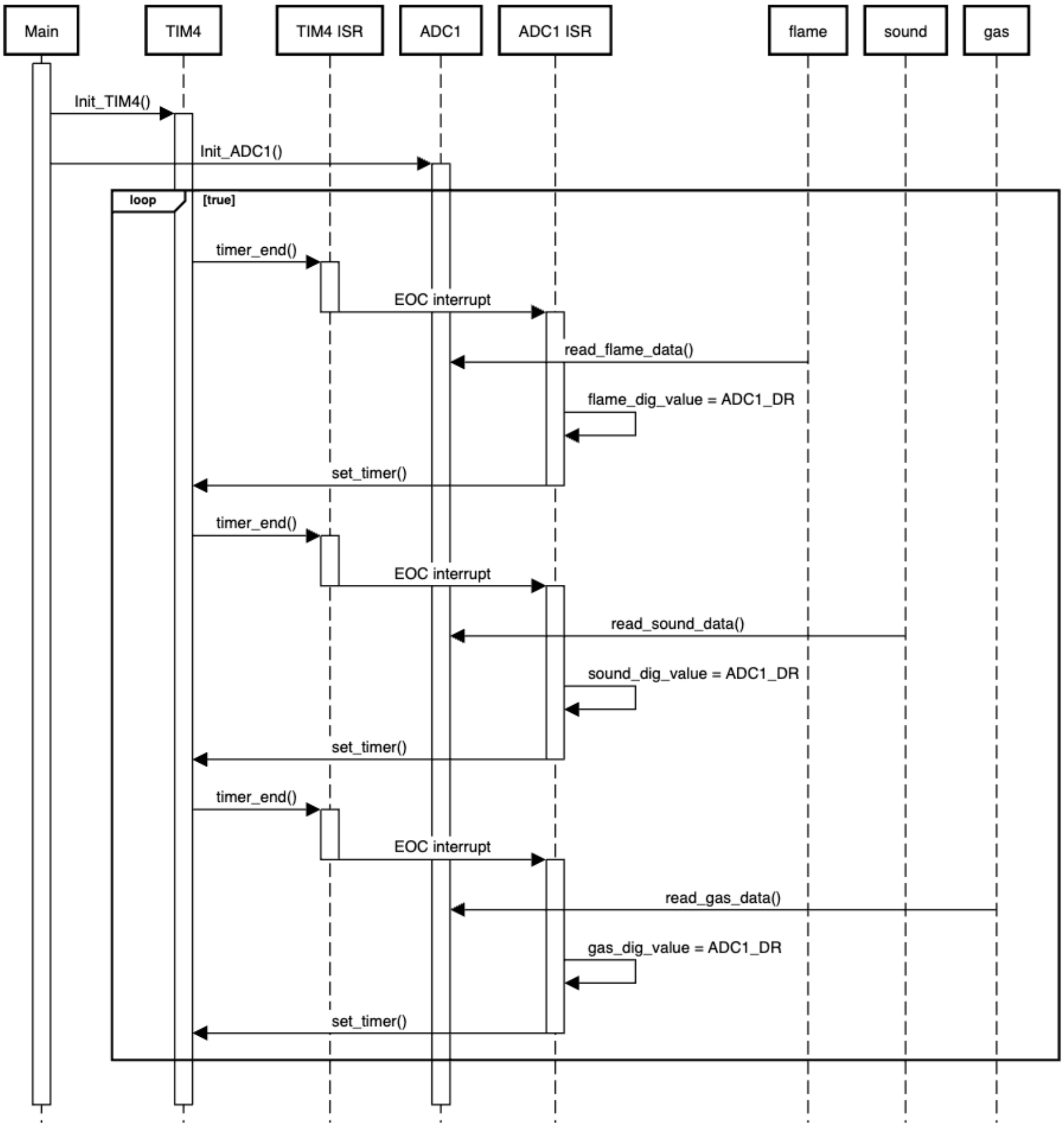


Figure 2.4: Sequence Diagram for ADC

Algorithm 6 `UART Communication`

$inp = getInfo()$
if $inp == w$ **then**
  $sendWeightInfo()$
end if
if $inp == f$ **then**
  $sendFulnessInfo()$
end if

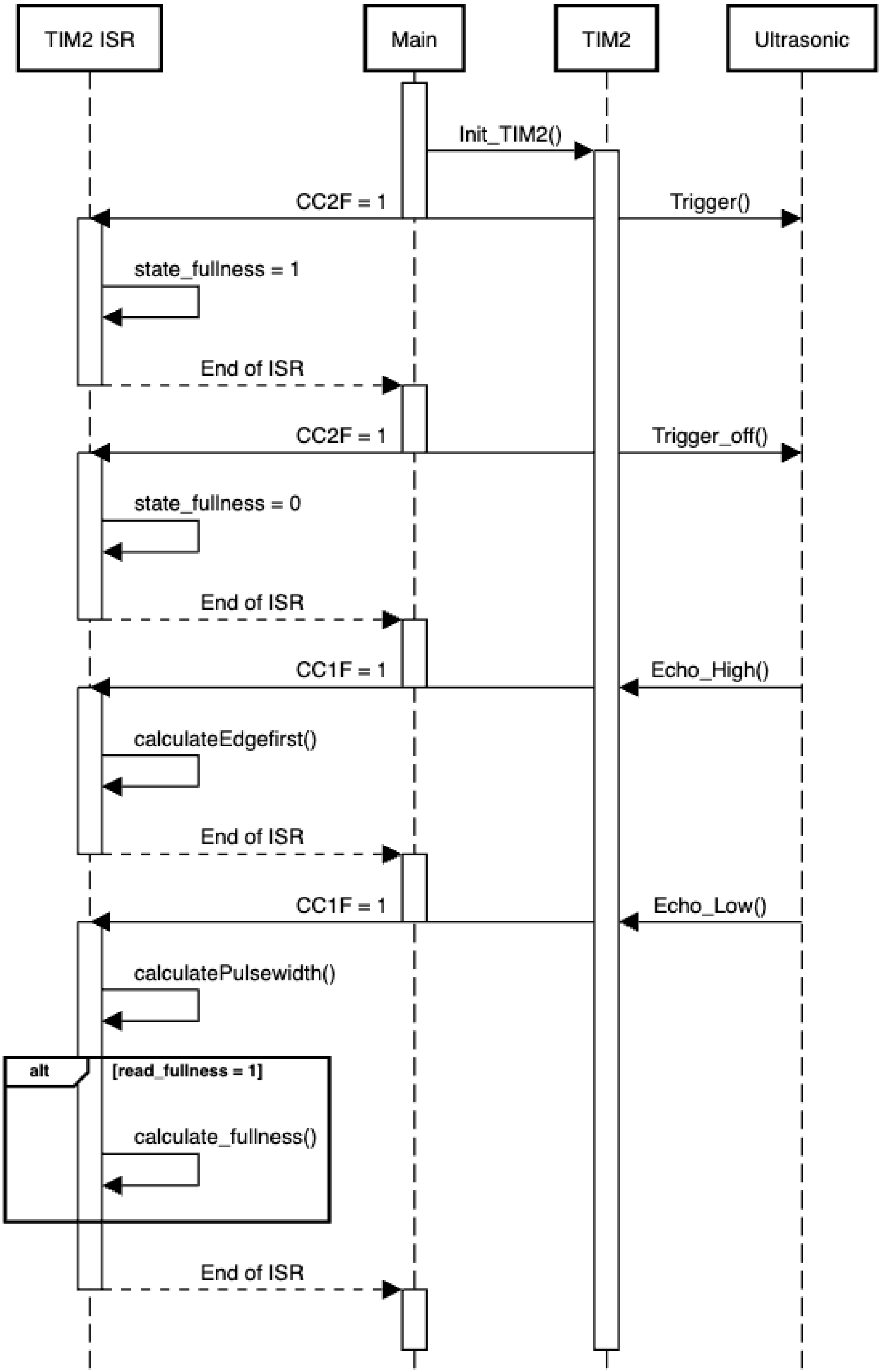


Figure 2.5: Sequence Diagram for Calculating Fullness

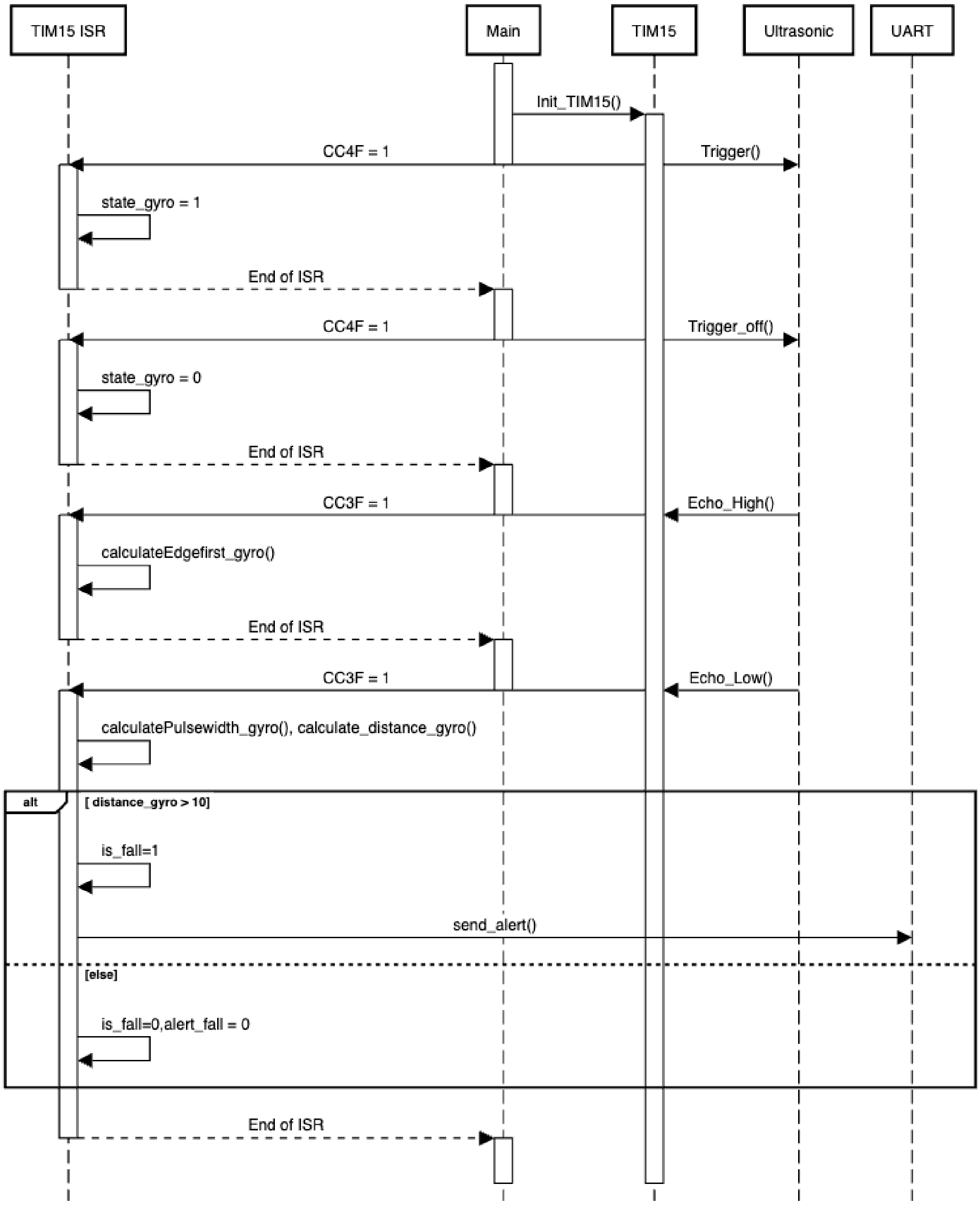


Figure 2.6: Sequence Diagram for Fall Functionality

Algorithm 7 `Alert Sending Generic Algorithm`

if $isBiggerThanthreshold(sensorValue)$ **then**
  if $checkAlertOff(sensor)$ **then**
    $sendAlert()$
    $setAlertOn(sensor)$
  end if
else
  $setAlertOff(sensor)$
end if

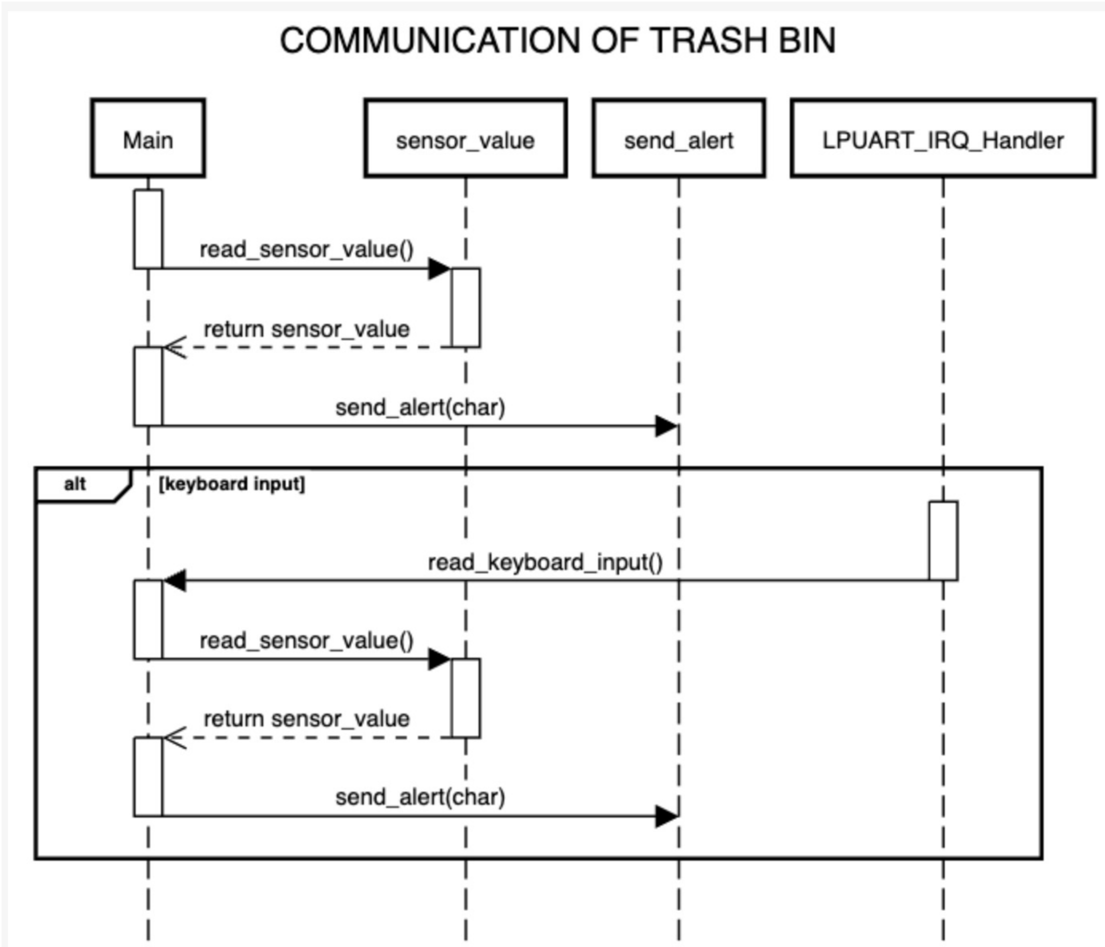
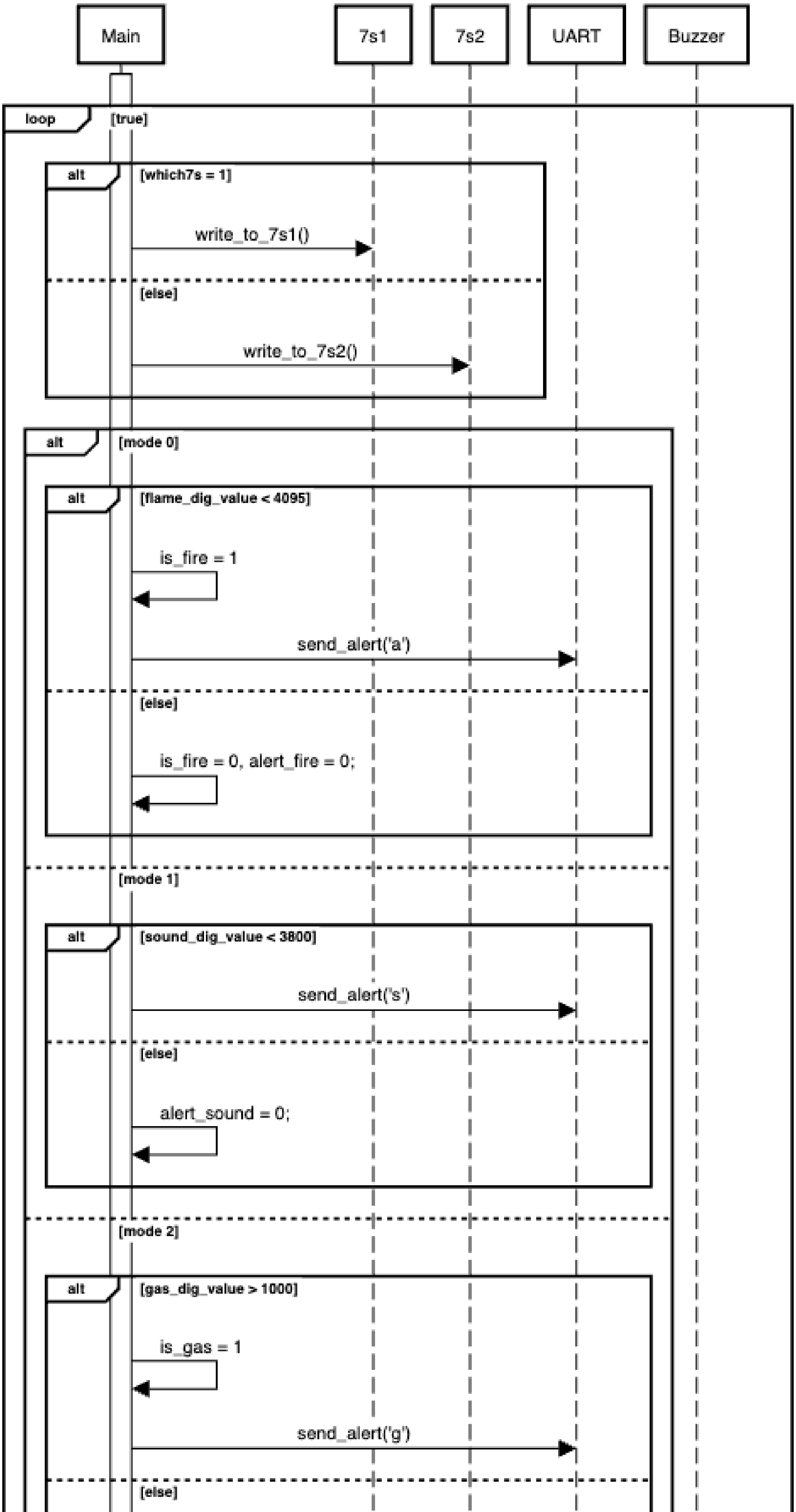
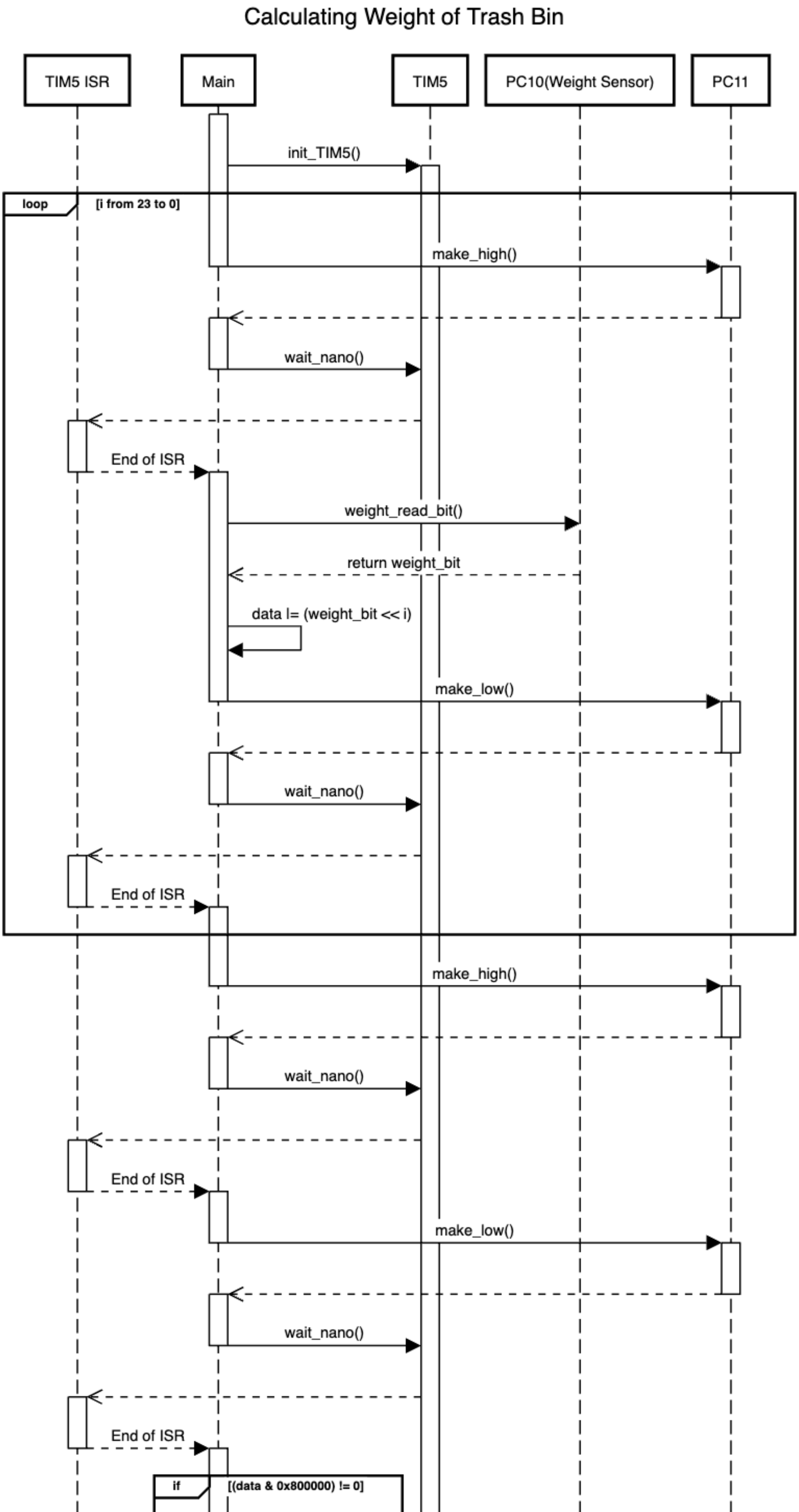


Figure 2.7: UART Diagram For Communication

Algorithm 8 `Height Detection`

if $CCR2interrupt()$ **then**
  if $state == 0$ **then**
    $setCCR2TOn()$
  else
    $setCCR2TOff()$
  end if
end if
if $CCR1interrupt()$ **then**
  if $icpin == 0$ **then**
    $calculateEdgefirst()$
  else
    $calculatePulsewidth()$
    $percentage = calculateFullness()$
    $Write7Segment1(percentage)$
    $Write7Segment2(percentage)$
  end if
end if


Main
7s1
7s2
UART
Buzzer
loop
[true]
alt
[which7s = 1]
write_to_7s1()
[else]
write_to_7s2()
alt
[mode 0]
alt
[flame_dig_value < 4095]
is_fire = 1
send_alert('a')
[else]
is_fire = 0, alert_fire = 0;
[mode 1]
alt
[sound_dig_value < 3800]
send_alert('s')
[else]
alert_sound = 0;
[mode 2]
alt
[gas_dig_value > 1000]
is_gas = 1
send_alert('g')
[else]



Calculating Weight of Trash Bin
TIM5 ISR
Main
TIM5
PC10(Weight Sensor)
PC11
init_TIM5()
loop
[i from 23 to 0]
make_high()
wait_nano()
End of ISR
weight_read_bit()
return weight_bit
data |= (weight_bit << i)
make_low()
wait_nano()
End of ISR
make_high()
wait_nano()
End of ISR
make_low()
wait_nano()
End of ISR
if
[(data & 0x800000) != 0]

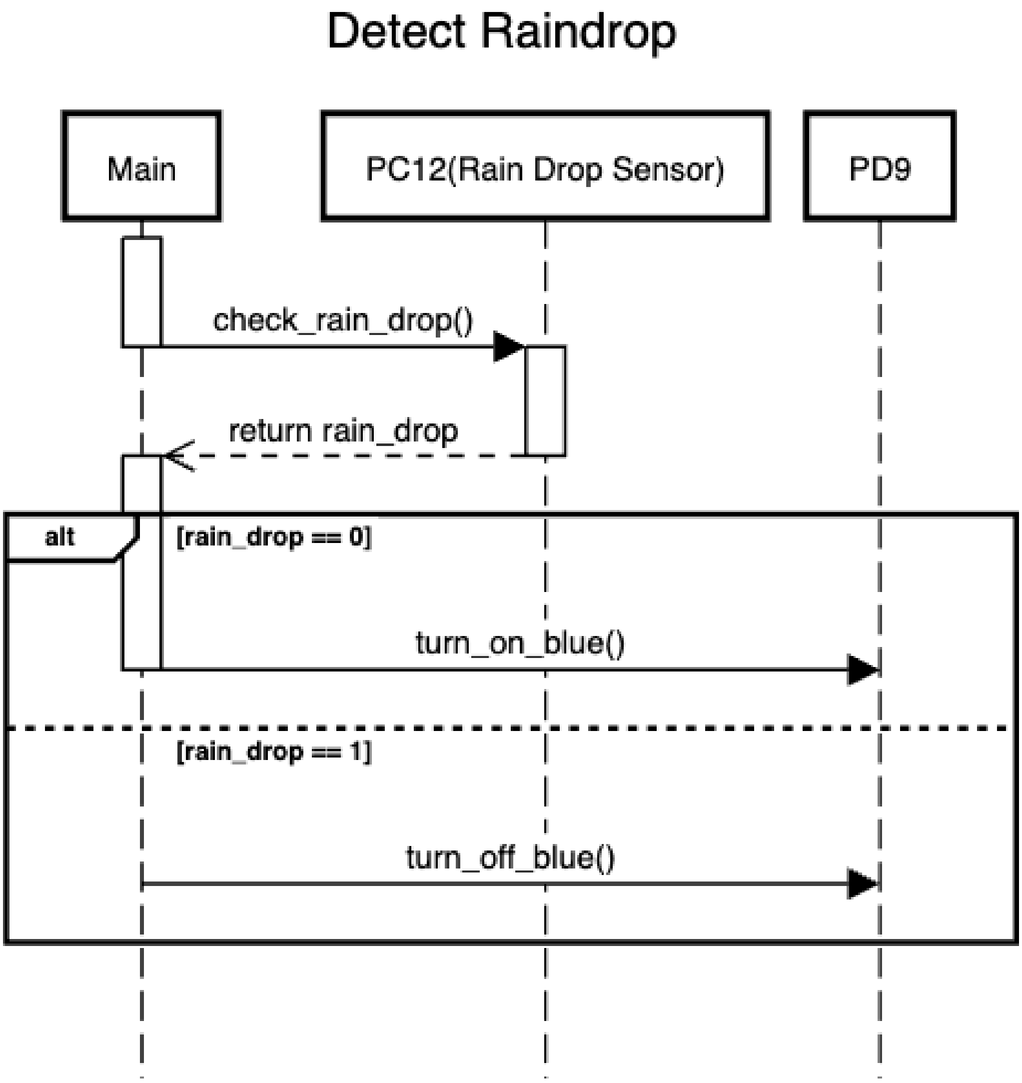


Figure 2.10: Sequence Diagram for RainDrop

Algorithm 9 `Automatic Lid Functionality`

if $CCR2interrupt()$ **then**
  if $state == 0$ **then**
    $setCCR2TOn()$
  else
    $setCCR2TOff()$
  end if
end if
if $CCR1interrupt()$ **then**
  if $icpin == 0$ **then**
    $calculateEdgefirst()$
  else
    $calculatePulsewidth()$
    $distance = calculateDistance()$
    if $distance < 8$ **then**
      $openLid()$
    else
      $wait3Second()$
      $closeLid()$
    end if
  end if
end if

2.4 Timing Diagram

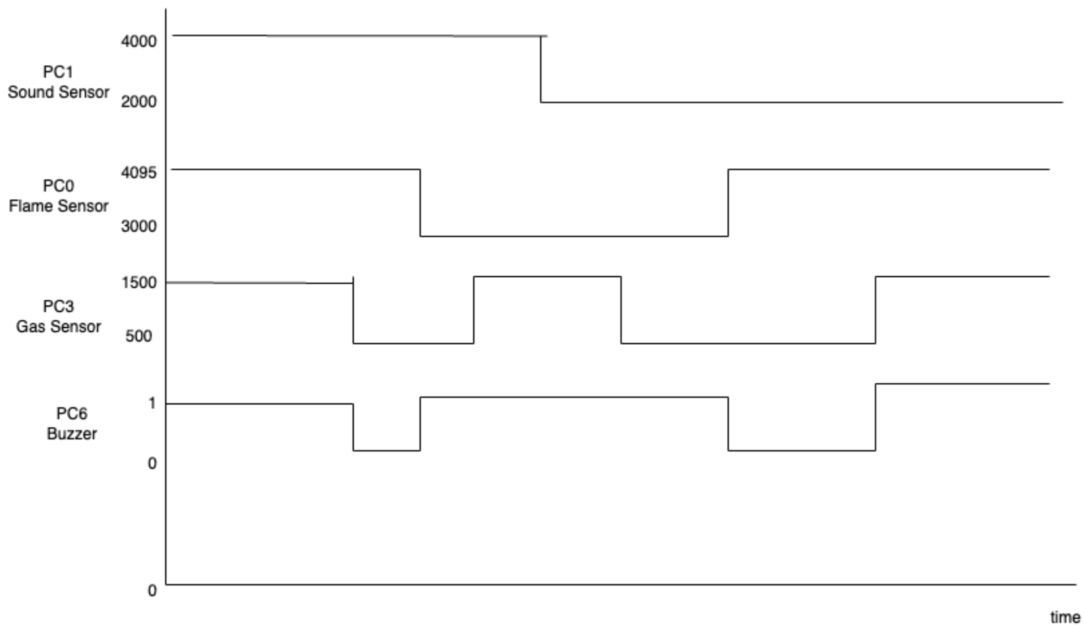


Figure 2.11: Timing Diagram for ADC

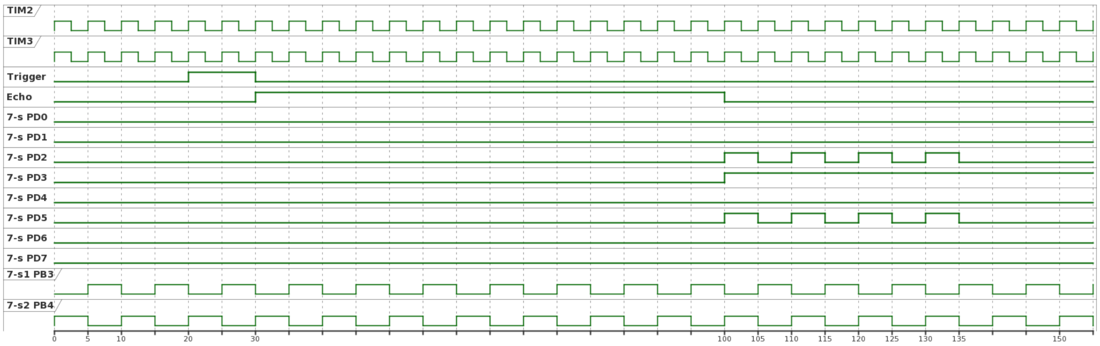


Figure 2.12: Timing Diagram for Ultrasonic Fullness

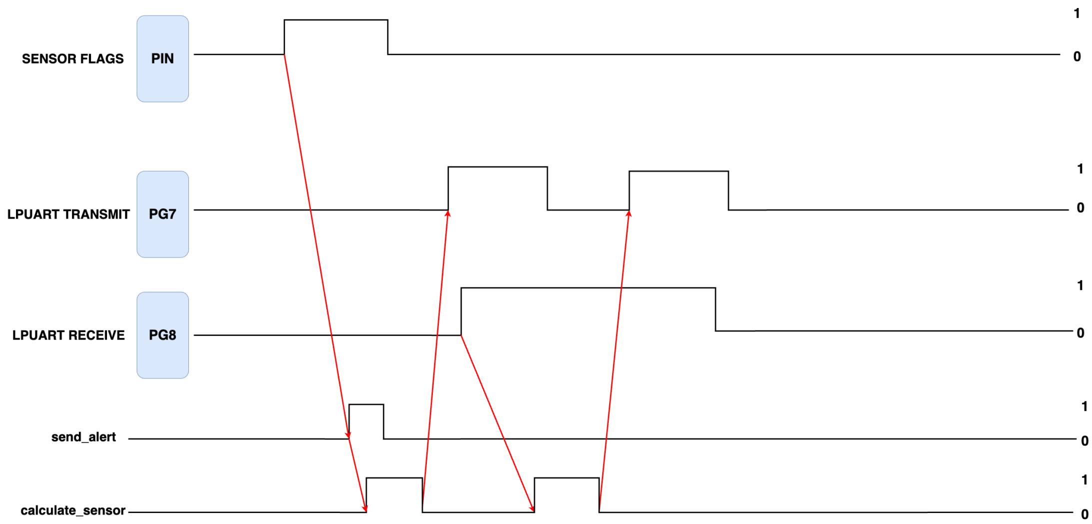


Figure 2.13: Timing Diagram for Uart Communication

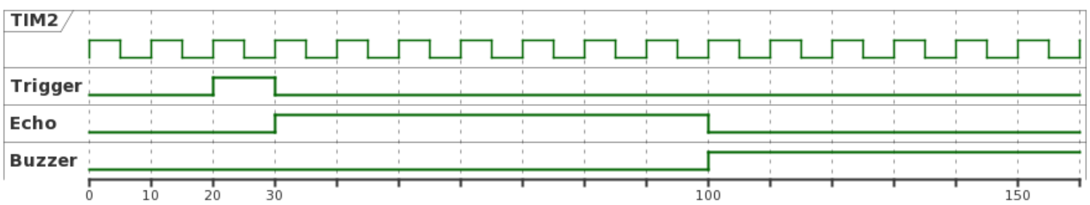


Figure 2.14: Timing Diagram for falling

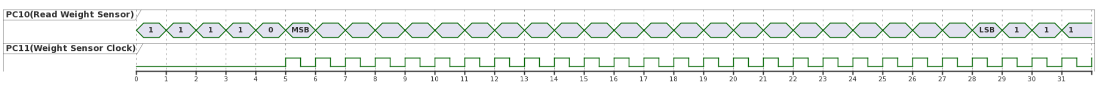


Figure 2.15: Timing Diagram for Weight

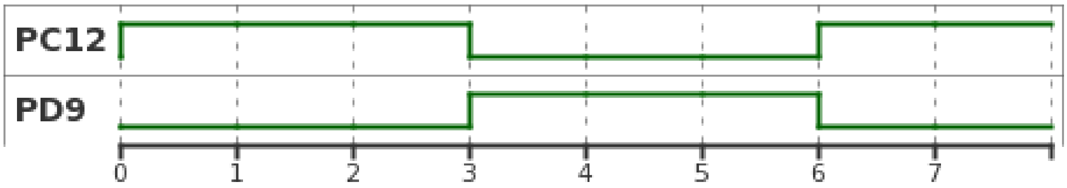


Figure 2.16: Timing Diagram for Rain Drop

3. Discussion and Experiential Evaluation

- **Usefulness of Our Design**

  As stated in our conceptual design proposal, our design is helpful in many scenarios we encounter in our daily lives. For example, with the help of our sensors and wired communication protocol, users of our smart trash bin will be informed rapidly. In a more concrete way, users will be informed in situations involving fire and dangerous/toxic gases. In order to understand the usefulness of this phenomenon one can think of hundreds of trash bins located in a shopping mall. By adding id (for further advancement of the prototype) of the corresponding trash bin, attendants in charge may handle the dangerous situation occurred inside the trash bin such as situations stated above. Without going far away from our shopping mall example, we also want to mention our additional functionality that can be useful on liquid leakage scenarios. Sometimes, we encounter issues with our household trash bins, possibly because we overload them with garbage or include sharp objects. This may result in tearing of trash bags and the accumulation of undesired liquids at the bottom of the bin. Those liquids tend to have a bad odor that affects the overall air quality of the surrounding environment. So think of this liquid leakage scenario in a mall that consists hundred of trash bins. This kind of leakage scenarios may turn every visitor's experience into an unbearable one. Via our liquid detector/rain drop sensor, attendants can easily find the trash bin that results in some sort of a leakage and clean it rapidly.

  As an additional feature we may propose that increase the functionality and usefulness of our design is a similar design idea that we encounter in almost every location that requires hygiene such as restrooms which is touchless open/close functionality that can be applied to doors, toiler seat covers, etc. We've implemented this touchless opening/closing idea for our trash bin's lid. Users can open the lid of our trash bin only using motion/gesture near to the lid. Motion will trigger the ultrasonic sensor we've implemented and that ultrasonic will sequentially trigger the servo motor located next to the lid that will handle the required actions. This implementation of touchless lid action simply serves for increased hygiene and more convenient usage of our system. Additionally, our implementation also includes a fall detection mechanism, that informs the nearby users with an additional buzzer and informs the admin-like users (e.g. attendants at a shopping mall) via via wired communication protocol. Similar to previously stated features, attendants can rapidly be informed about the tripped over trash bin and handle the situation gracefully without any scattered garbage or unwanted liquid that resides on the floor. Although we give concrete examples as if we designed our trash bin specifically to shopping malls, same reason - solution relationship can be constructed for household trash bins and garbage dumpsters on the streets.

To sum up, our smart trash bin design proposes an innovative and convenient functionality compared to traditional trash bins which are simply a serves as storage areas for trash. Beside its features that increase the overall user experience, with the help of sensors and communication protocol, individuals can avoid undesired and dangerous scenarios that they may encounter in their daily lives.

- **Limitations of Our Design**

  First limitation that comes to mind after one's evaluation of our design may be the communication protocol used throughout the workflow of the system. Since a wired communication is used, the alerting logic mandates the user to be near to the system unless a complicated wiring system (e.g. lots of appended USB cables) is decided to be used. Another limitation of our design we can mention is lack of user-friendliness off indicators. We've implemented 7-segment displays to inform users about the current fullness of the trash bin. We might have come up with a different displaying component such as 16x2 LCD display which would provide a more readable and understandable user interface for our design. As a final limitation of our design, we can think of the overall power consumption of the system. Since we are using numbers of sensors and they are constantly checking for a new update during the workflow, although we make some of them to sleep for a percentage of unit time, they consume non-negligible amount of power overall.

- **How Can We Improve Our Design ?**

  As we've stated in previous section where we mention about limitations of our design, wired communication protocol may be considered as one of the key downsides of our design. In order to improve our design we could have used a wireless communication protocol between a machine (another microcontroller or a computer) and our microcontroller board. As a more concrete design idea, we could have used a bluetooth module or another kind of wireless communication module (e.g. WiFi) to enable users to act freely while communicating with microcontroller. Which is objectively a more convenient communication protocol compared to basic wired protocols. Another improvement we could have include in our design is adding GPS-like functionality to trash bins. Which would be quite useful for locating the specific trash bins. This feature would potentially be more beneficial for shopping malls and garbage bins on streets since hundreds of trash bins are located along shopping malls (and/or on streets). Via using GPS information, we might have implemented shortest path and/or minimum spanning tree algorithms to find the shortest path to a specific trash bin and a path to collect/check each trash bin on an area in shortest time possible, respectively.

- **Our Experience Throughout Design**

  First of all, the technical difficulties we've encountered were mostly expected issues we had on our mind even before starting implementing our prototype. More concretely, every team member needed to do his part of the project (if there isn't any blocking situation among members) on his own. Hence, every member needed to check the data sheet of the component he uses or read the relevant documentation for the corresponding functionality from the files provided in the syllabus of the course. Secondly, although everyone did his job individually, we needed to combine each logic that is used for each component and peripheral into a single microcontroller board. So in terms of technical difficulties we can count individual work (using the component/feature each member is assigned to) and team work (combining every component feature to work all together into a single microcontroller board) separately.

  We can say that, we didn't come accross with any social difficulties. Each member luckily knows each other pretty well, so everyone freely presented his ideas about almost every part of the implementation and design of the prototype. We've had meetings regularly, which allowed us to plan our path throughout development process.

  As a team, we think that we did a pretty good job implementing every topic we've covered during lectures and practiced during lab hours. We extended both our practical knowledge and skills via adding more features to our system such as additional sensors and peripherals. Additionally, we think that combining every individually implemented feature into a single project enhanced our understanding on embedded system programming and bare metal coding overall.

References

[1] A. Hernández-García and P. König, “Data augmentation instead of explicit regularization,” *arXiv preprint arXiv:1806.03852*, 2018.