

DEVELOPMENT OF DROWNING PREVENTION SYSTEM FOR CHILDREN LEARNING IN SWIMMING POOL

Submitted in partial fulfillment of the requirements for the award of

Bachelor of Engineering Degree in
Electrical and Electronics Engineering
by

RADHA M (Register No. 38140040)

ANITHA N (Register No. 38140007)



**DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING
SCHOOL OF ELECTRICAL AND ELECTRONICS ENGINEERING**

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

Accredited with Grade "A" by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119

APRIL 2022



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **RADHA M** (Reg. No. 38140040) and **ANITHA N** (Reg. No. 38140007) who carried out the project entitled "**DEVELOPMENT OF DROWNING PREVENTION SYSTEM FOR CHILDREN LEARNING IN SWIMMING POOL**" under our supervision from December 2021 to April 2022.

Dr. M Pushpavalli
Internal Guide

Dr. V. SIVACHIDAMBARANATHAN
Head of the Department

Submitted for Viva voce Examination held on _____

Name:

Signature:

Internal Examiner

External Examiner

DECLARATION

We **RADHA M** (Reg. No. 38140040) and **ANITHA N** (Reg. No. 38140007) hereby declare that the Project Report entitled “**DEVELOPMENT OF DROWNING PREVENTION SYSTEM FOR CHILDREN LEARNING IN SWIMMING POOL**” done by us under the guidance of **Dr. M Pushpavalli** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Electrical and Electronics Engineering.

DATE: 20.04.2022

PLACE: CHENNAI

SIGNATURE OF THE CANDIDATES

1. 
2. 

ACKNOWLEDGEMENT

We are pleased to acknowledge our sincere thanks to the Board of Management of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. We are grateful to them.

We convey our thanks to **Dr. N. M. Nanditha., Dean, School of Electrical and Electronics Engineering** and **Dr. V. Sivachidambaranathan, Head of the Department, Dept. of Electrical and Electronics Engineering** for providing us necessary support and details at the right time during the progressive reviews.

We would like to express our sincere and deep sense of gratitude to our Project Guide **Dr.M Pushpavalli** Assistant professor for her valuable guidance, suggestions and constant encouragement paved the way for the successful completion of our project work.

We wish to express our thanks to all Teaching and Non-teaching staff members of the Department of Electrical and Electronics Engineering who were helpful in many ways for the completion of the project.

ABSTRACT

Every year, many individuals, including kids under the age of 5 drown in the deeps of the swimming pool, and the lifeguards are not well trained enough to handle these situations. Thus, arises the requirement for having a system that will consequently detect the drowning individuals and a voice alert life guard at such risk. Swimming pool surveillance system plays an specific role in safeguarding the premises. A system for preventing infant drowning is strongly required. The prototype of infant drowning prevention system was presented. The project developed on android software and hardware system. This Project, differential conductivity sensor is used for detection of drowning incidents In swimming pools at the earliest possible stage. The children's life is saved during drowning incidents in the swimming pool. This proposed approach consists of conductivity Sensor, Nano microcontroller, LCD display, HC-05 module , relay, oxygen pump, buzzer and android app . The demo system based on conductivity sensor has an advantage of convenience, cost saving and simple algorithm . This system monitors vital parameters such as drowning condition, movement of an infant. The GCM network this information is transferred to their parents. Measurements of this vital parameters can be done and under risk situation conveyed to the caretaker with alarm triggering system to initiate the proper control actions.

TABLE OF CONTENTS

CH. NO	TITLE	PAGE NO
	ABSTRACT	v
	LIST OF FIGURES	ix
1.	Introduction	
	1.1.Overview	1
	1.2.Objective	2
	1.3.Problem Statement	2
2.	Literature Survey	3
3.	Aim and Scope of the present investigation	
	3.1.Existing system	6
	3.2.Block diagram of existing system	6
	3.3.Proposed System	7
4.	Simulation Analysis	
	4.1 Introduction to Embedded C	9
	4.2 Keil software	9
	4.3 Simulation results	10

5.	Hardware	
5.1	Block diagram of hardware system	12
5.2	Circuit diagram in normal condition	13
5.3	Circuit diagram in drowning condition	14
5.4	Circuit diagram	15
5.5	Nano microcontroller	16
5.6	Power supply	17
	5.6.1 IC Voltage Regulator	18
	5.6.2 Three-terminal Voltage Regulator	19
5.7	ADC	19
5.8	Conductivity sensor	19
5.9	Bluetooth	20
5.10	Buzzer	21
5.11	LCD display	22
5.12	Relay	22
5.13	Oxygen Pump	23
5.14	Android Application	23
5.15	Java	24
5.16	Block diagram of android software	25

	5.17	Process flowchart for android app	26
	5.18	Android software results	27
6.	6.1	Conclusion	31
	6.2	Future scope	31
		Appendix source code	35

LIST OF FIGURES

FIG.NO	TITLE	PAGE.NO
3.1	Block diagram of existing system	6
3.2	Block diagram of proposed system	8
4.1	Simulation results under normal condition	10
4.2	Simulation results under drowning condition	11
5.1	Block diagram of hardware system	12
5.2	Hardware circuit under normal condition	13
5.3	Hardware circuit under drowning condition	14
5.4	Circuit diagram	15
5.5	Pin diagram of nano microcontroller	17
5.6	Power supply circuit diagram	18
5.7	HC-05 Bluetooth Module	21
5.8	Oxygen pump	23
5.9	Block diagram of android software	25
5.10	Process flowchart	26
5.11	Home page of an android software	27
5.12	Select user mode page of android application	27
5.13	New trainer details page	28
5.14	Registered trainer details page	28

5.15	Student details page	29
5.16	Drowning detection page	29

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

The World Health Organization (WHO) outline the drowning as a method of experiencing metabolism impairment result of submersion or immersion within the liquid like water. The blockage of the human system might suffer the respiratory organ and therefore the brain (hypoxia) thanks to lack of gas flow within the human blood. Usually, drowning happened in open house like streams, lakes, bathtubs and natatorium. Lack swimming talents among the victims and fewer effective strategies of the water safety watching round the natatorium area unit the key purpose for this example. The wide space of natatorium and congestion of holiday makers at the natatorium might produce several blind spots. The lifeguard's read of the swimming space is obscured. it's the tender's responsibility to hunt out each blind spot within the swimming space and to require the required action to scale back the quantity of space that the tender cannot see from his/her lifeguard station. it would offer difficulties for lifeguards to watch all the swimmer's safety within the natatorium. the prospect of drowning among the youngsters is high are the end result of this issue. Swimming could be a favorite exercise, however within the water, beginners usually can't be liberal to breathe action and cause choking water, then loss of balance and cause a drowning accident.

Some special circumstances, like cramps, impinge on one another, malady or mental stress, and so on, can also cause swimmer drowning. Drowning could be a leading reason for death and incapacity for youngsters. Worldwide, drowning produces the next rate of mortality than the other reason for injury in kids but fifteen years old-time. Young kids below the age of 5 area unit at specific risk, suffering the

very best drowning mortality rates worldwide. Young kids below the age of 5 area unit at specific risk, suffering the very best drowning mortality rates worldwide. during this project to avoid drowning accidents mechanically by mistreatment the physical phenomenon plate unendingly measured and monitored by the microcontroller. once the present price surpasses the edge limit associate alerting signal is distributed to the receiver. The wireless transmission and reception of signals is finished through Bluetooth module. On receiving the valid signal microcontroller sets the buzzer ON and SMS notification can send to oldsters by mistreatment GCM communication.

1.2 OBJECTIVE

- In this project to avoid drowning accidents by using conductivity sensor and wireless communication.
- This device continuously measured and monitored by the controller in control room.
- Our device judge whether the human body in the water, if the body is drowning in the water 10 seconds the condition will be conformed and the information send to control room with the help of HC-05 wireless device.
- By using this project, we safeguard the human life easily with proper communication.

1.3 PROBLEM STATEMENT

Drowning is a leading cause of death and disability for children. Worldwide, drowning produces a higher rate of mortality than any other cause of injury in for children less than 15 years of age. In this project, to avoid the drowning accidents automatically detected by using the iron plate continuously measured, monitored by the microcontroller.

CHAPTER 2

LITERATURE SURVEY

2.1 LITERATURE SURVEY

Chaudhari, T et.al(2018)proposed the idea to save a person's life who is drowning in the water by informing heart rate to the lifeguard using a transmitter and receiver RF module. The transmitter unit consists of heart rate sensor, LCD, switch and regulator connected to the microcontroller powered by a 12V battery. The heart rate sensor is mounted on the hand or the head of the swimmer to track the person's heart rate. The LED and buzzer will be turn on if the heart rate of the swimmer increased or decreased. The receiver unit consist of a RF module, buzzer, microcontroller, transformer, rectifier and regulator will be hold by the lifeguard. Upon the detection of a drowning symptom, the buzzer will be ON to alert the lifeguard.

AlMahmoud, et.al (2018)depicts the methodology inrecent development in the field of IoT have led to a renewed interest in using sensors to identify and alert the event of drowning. Al Mahmoud developed a device composed of sonar sensors to identify a drowning person inside a body of water to form a three-dimensional sonar wave grid within the water body. A microcontroller is connected to the sensors for tracing a movement path of a person. The system will analyze the movement path of the person. An alarm will be issued if the movement path is indicative of a drowning person .

Ramani, J. G., et.al (2019)proposed the idea of measure and record the normal pulse pressure of the swimmer before they were made to wear the wristband watch and enter the pool. A low and high threshold is set to account for variation in the heartbeat, suggesting that the swimmer began to drown. The pulse rate is transmitted

through the RF module. The Arduino Uno microcontroller will have received the signal and triggers an alarm signal through the buzzer. The microcontroller will also display an emergency text on the LCD in order to get the attention of the lifeguard for help to the swimmer. A similar wristband system has been proposed but a GSM module is used to communicate to the receiving unit.

A.Roy, et.al (2018)proposed wearable swimming goggles with a hydrophone, a buzzer is in outside water pool. Hydrophones were sending alert to buzzer outside water. Implemented using Proteus software and demonstrated practically. Hydrophones sound may be jammed by noise in public pools, and discomfort to the user to wear a mask type goggle.

Ramani, J. G.,et.al(2019)measure and record the normal pulse pressure of the swimmer before they were made to wear the wristband watch and enter the pool. A low and high threshold is set to account for variation in the heartbeat, suggesting that the swimmer began to drown. The pulse rate is transmitted through the RF module. The Arduino Uno microcontroller will have received the signal and triggers an alarm signal through the buzzer. The microcontroller will also display an emergency text on the LCD in order to get the attention of the lifeguard for help to the swimmer. A similar wristband system has been proposed but a GSM module is used to communicate to the receiving unit.

Carballo-Fazanes, et.al(2019)studied and analyzed the visible behavior of drowning persons, using drowning videos. These videos are observed and analyzed by both international water-safety experts and the Lince observation software to identify drowning persons and provide early behavioral detection.

Castaneda, et.al (2018)describes thePPG/Heart beat sensor are functional through two main components which are the LED and photodiode. Both of the component are placed next to each other and along with circuit.

Carballo-Fazanes, et.al(2019)proposed the idea ofPPG/Heart beat sensor are functional through two main components which are the LED and photodiode. Both of the component is placed next to each other and along with circuit.

Chaudhari, et.al(2019)Based on studies, their projects do not involve any Wi-Fi to connect between the microcontroller to get an alert notification through mobile phone. Wi-Fi technology enable multiple devices to be connected simultaneously. This will enable multiple wristband to be connected to the mobile phone which the previous work used one-to-one connection. Mobile phone is widely use and always be kept close to the person. Thus, this will enable parents to get a quick alert notifications and information of their children in the swimming pool where lifeguard is distracted or unavailable.

Liu, Wen,et.al (2019)proposed the idea in smart application based on IoT and ML can be designed to provide an intelligent swimming pool management system because most swimming pool management is based on the traditional method of hiring a lifeguard to observe the water level and drowning incidents. This is a challenging issue for swimming-pool owners with limited budgets.

CHAPTER 3

AIM AND SCOPE OF THE PRESENT INVESTIGATION

3.1 EXISTING SYSTEM

The wearable drowning monitor device can detect drowning accident and alarm. The device has four main modules, including microprocessor, power module, SD memory card module, LED warning module, water pressure sensor module. When swimming the human arm must constantly waving in the water, if drowning, arm motion of floating is significantly reduced, and if falling into the water, almost motionless. According to the physiological response of human drowning, it can detect drowning accident

3.2 BLOCK DIAGRAM

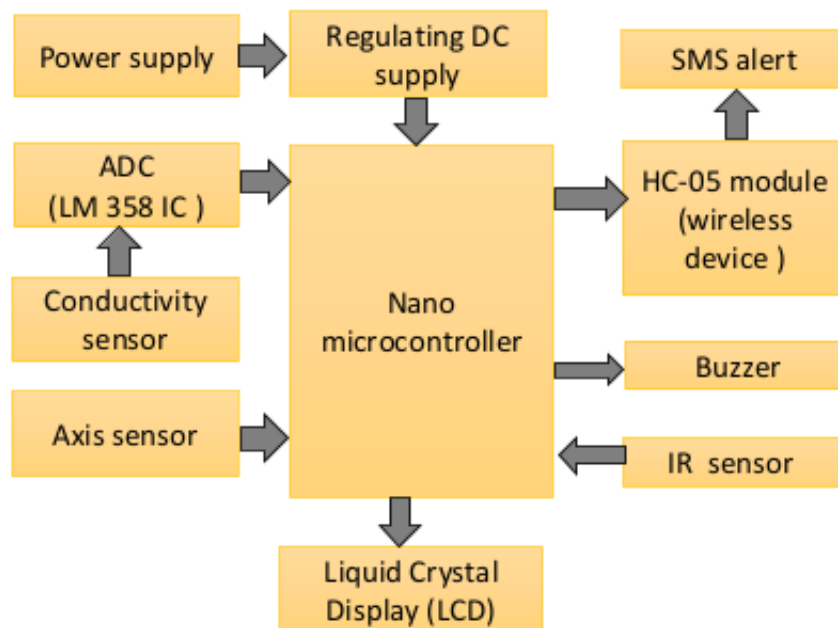


Fig 3.1 Existing System Block Diagram

Fig 3.1 represents the block diagram of existing drowning detection system

Disadvantages of Existing System

- Existing system detects the child drowning in the water and it raise the buzzer. But, it doesn't give oxygen until the child rescue from the drowning.

3.3 PROPOSED SYSTEM

In the projected methodology the kid drowning detection within the pool exploitation hardware devices and mechanical man application. First, information from a water conduction sensing element is employed to gauge whether or not the kid body within the water, if the body within the water, then begin downloading judgment method. Here we have a tendency to area unit employing a Nano microcontroller to regulate the complete method. Embedded c language is employed for the writing. once the sensing element price are going to be unbroken constant then the abnormally the alarm are going to be activated. The sensing element price are going to be modified with relevance the kid movement in water. If the drowning is detected the relay gets activated the chemical element pump. The SMS alert are going to be sent to the fogeys with the assistance of GCM communication through mechanical man application.

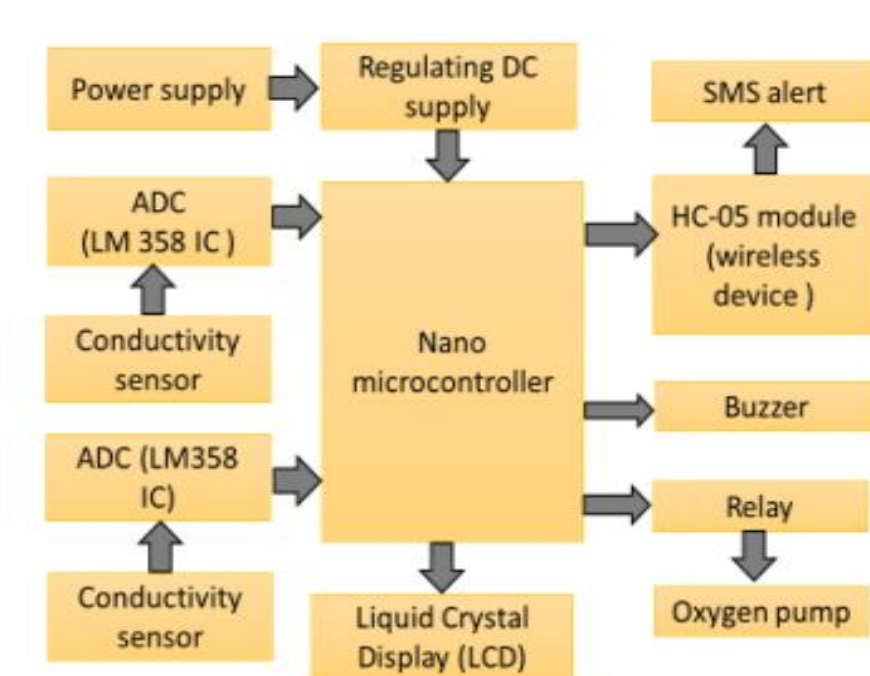


Fig 3.2 Proposed System block diagram

Fig 3.2 represents the block diagram of proposed system.

Advantages of Proposed System

- Our proposed system will detect whether the child body in water or not. If drowning is detected means it raises the alarm and send SMS to the guardian.
- Then the relay activates the oxygen pump and it gives oxygen to the child until the child rescue from drowning.

CHAPTER 4

SIMULATION ANALYSIS

4.1 INTRODUCTION TO EMBEDDED C

As programming language programs are specific to a processor, programming language didn't provide movable across systems. To beat this disadvantage, many high level languages, as well as C, came up. Other languages like PLM, Modula-2, Pascal, etc. conjointly came however couldn't notice wide acceptance. Amongst those, C got wide acceptance for not solely embedded systems, however conjointly for desktop applications. Even supposing C may need lost its refulgence as thought language for general purpose applications, it still has a strong-hold in embedded programming. Subsequent sections can discuss what Embedded C is, options of C language, similarities and distinction between C and embedded C, and options of embedded C programming. Compared to different high-level languages, C offers a lot of flexibility as a result of C is comparatively tiny, structured language; it supports low-level bit-wise information manipulation.

4.2 KEIL SOFTWARE

KEIL C software is used for microcontroller programming. C is efficient when compared to assembly language because, minimizes the lines of code - In assembly language, program which takes 100 lines will take 10 lines in Keil C.

- Easy to code and debug - C is easy to learn so it easy to code and since no of lines is less it will reduce complexity in debugging.
- Compatible with any microcontrollers - Just changing the header files we can make the program to work for different microcontrollers.

4.3 SIMULATION RESULTS

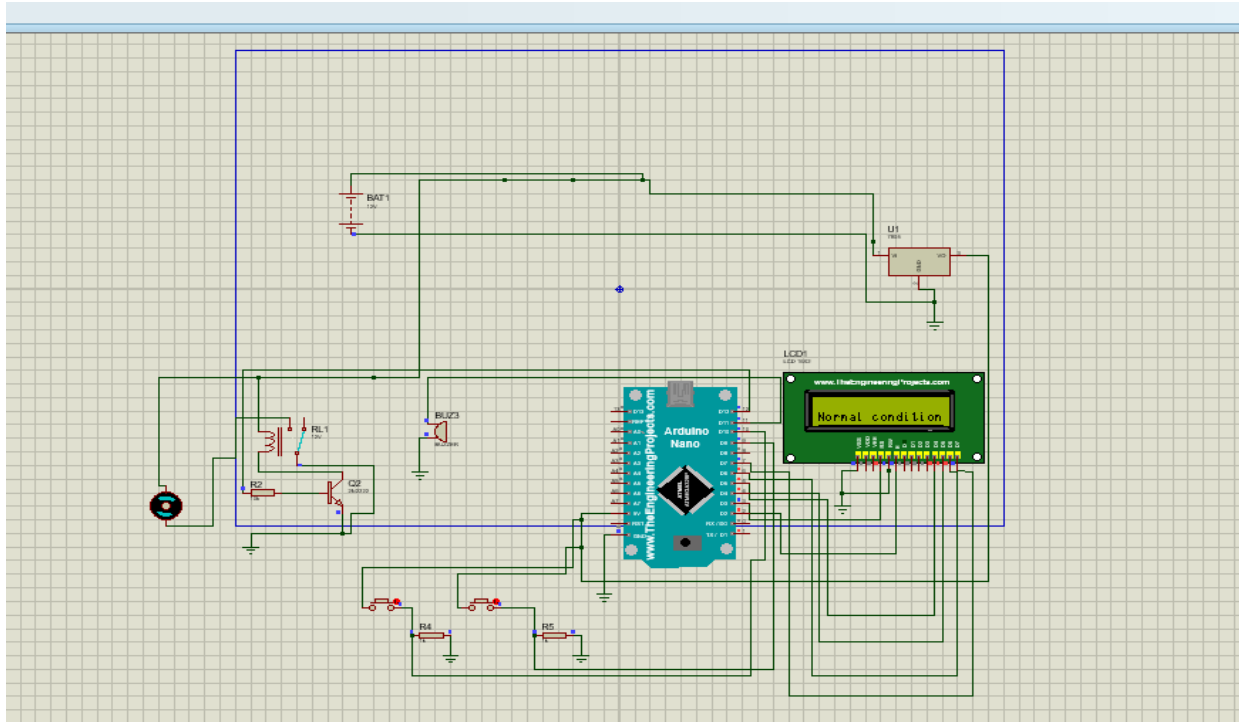


Fig 4.1 Represents the simulation at normal condition

This diagram represents when the child is under normal condition..in above fig switch is connected to Arduino microcontroller ,then microcontroller is connected to lcd .if the switch is closed means, Arduino nano microcontroller will send the information to lcd and it raises buzzer.here switch is opened ,so normal condition is detected and the condition displayed in LCD.

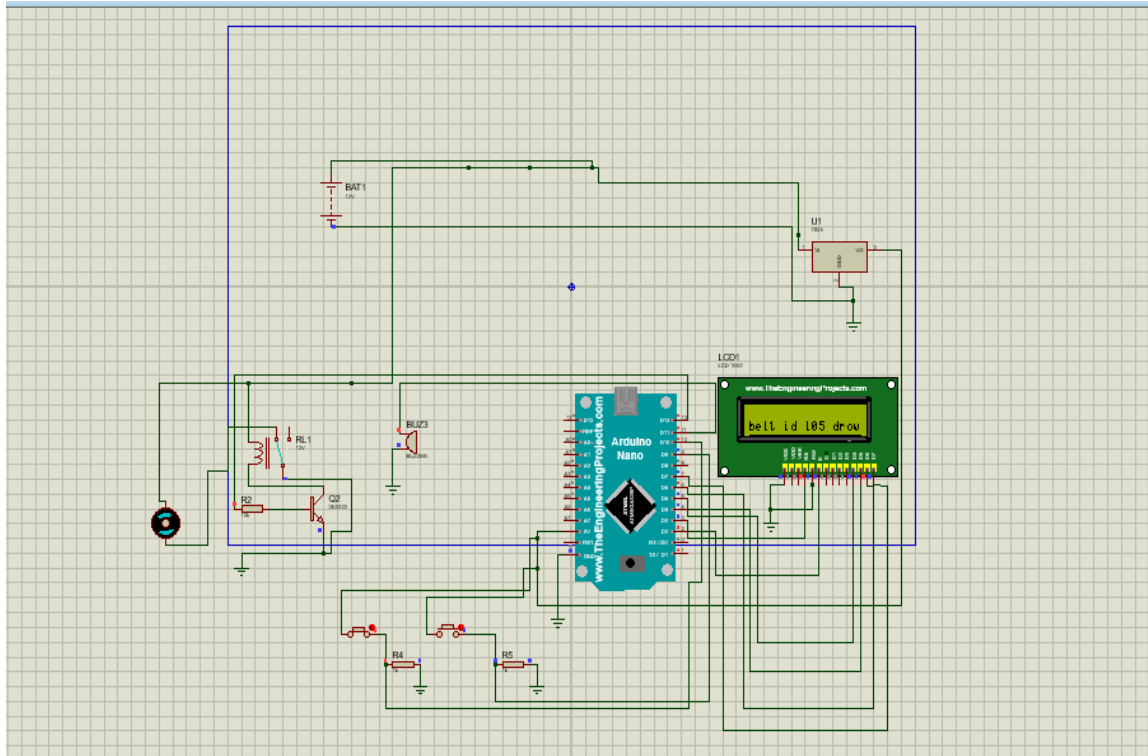


Fig 4.2 represents the simulation diagram when drowning is detected

In above fig switch is connected to Arduino microcontroller. the microcontroller is connected to lcd. the switch is closed so, then microcontroller sends the information to LCD. then lcd displays like child is under drowning condition. then, it raises the buzzer..

CHAPTER 5

HARDWARE

5.1 BLOCK DIAGRAM

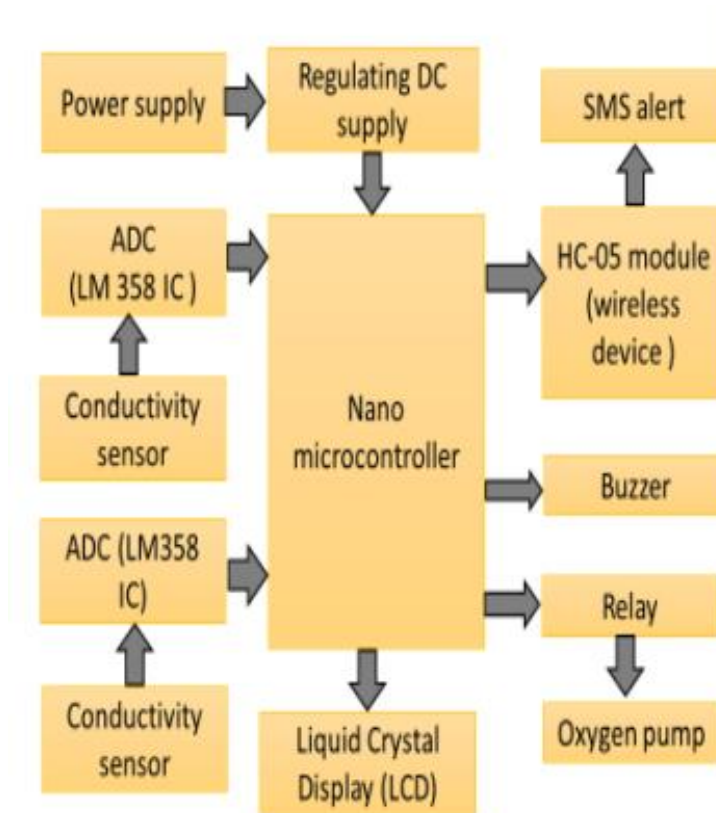


Fig 5.1 represents the block diagram of hardware system

The above block diagram 5.1 consisting the both hardware part and android application. The child drowning detection in swimming pool by this project. The hardware part consisting the microcontroller and sensors. The Nano controller will be controlling the whole device. Conductivity sensors are used to detect the child body in the water or not, the child can be monitoring continuously based on the sensor data the drowning is detected. If the drowning is detected means the buzzer is alarmed and the relay gets activate the oxygen pump to rescue the child. In software part the android application is developed. The app consists the new trainer,

registered trainer and student mode. They can use this application can enter the user id and password and if new trainer means fill the details. The sensor data from the hardware can be received in the android application by using the hc-05 wireless module. If the drowning is detected the SMS alert send to caretakers with the help of GCM communication.

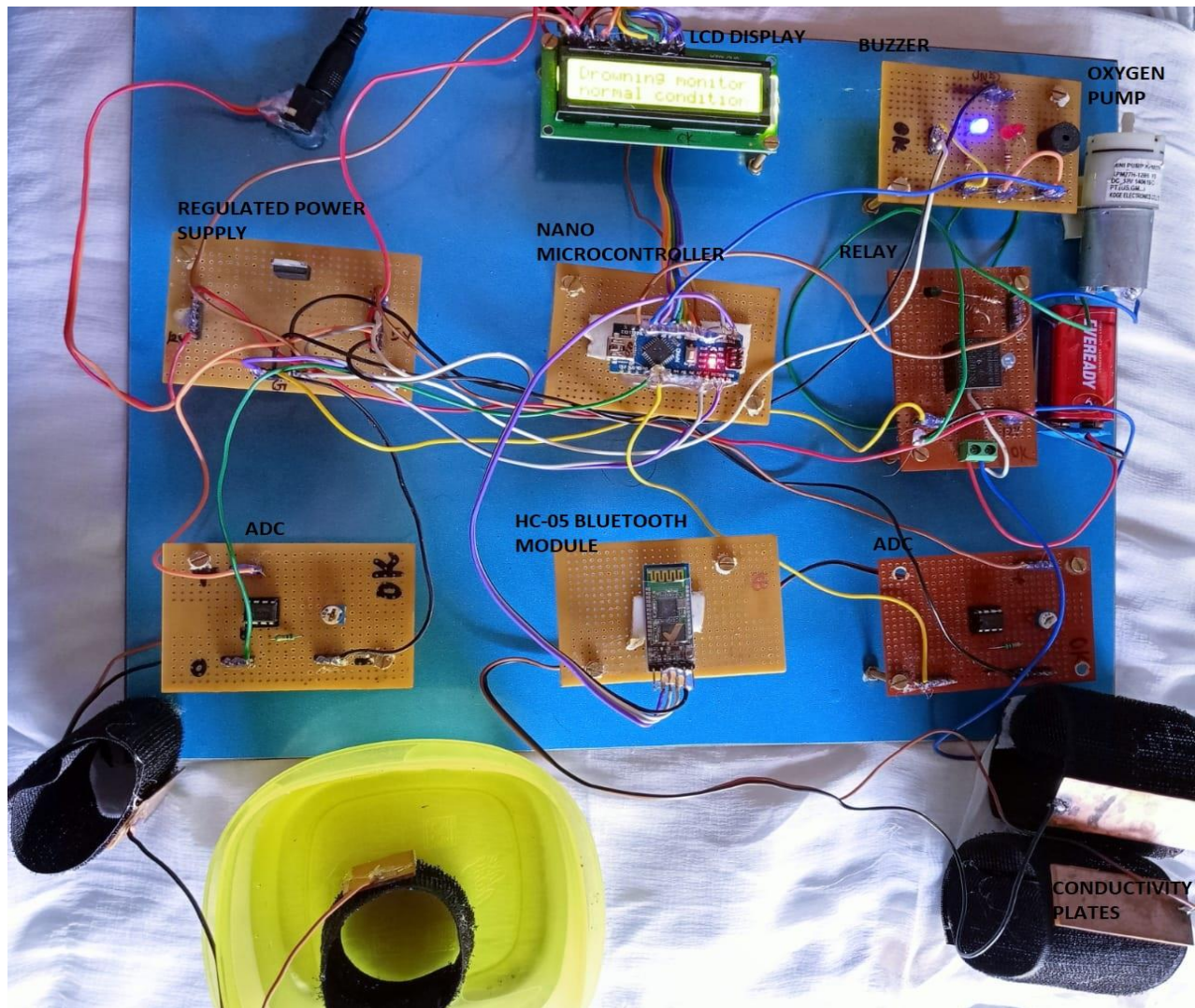


Fig 5.2 represents the hardware circuit at normal condition

In above 5.2 circuit, only one conductivity plate drowned in the water, so we can infer that the child is not drowned so oxygen pump and relay are not activated.

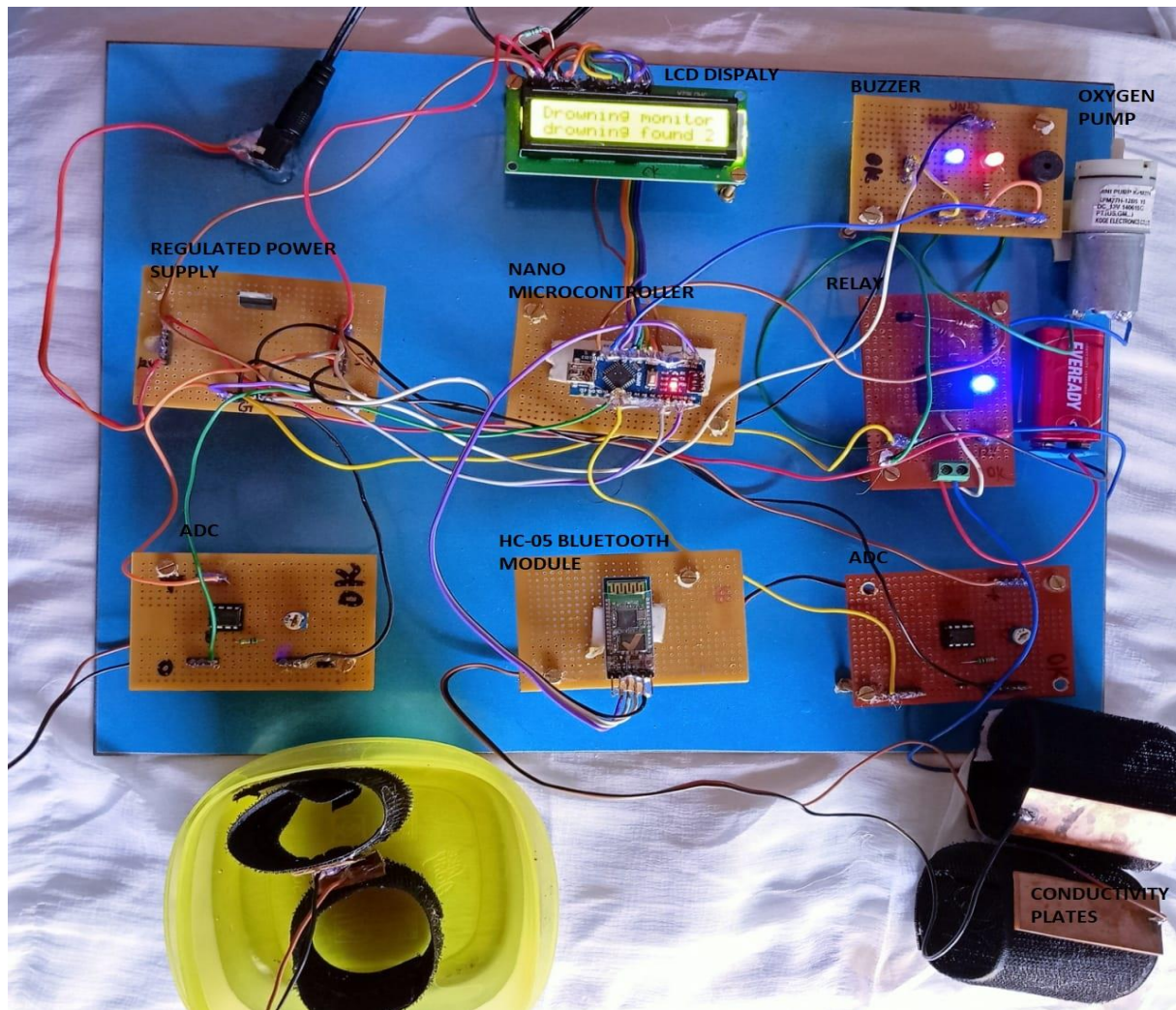


Fig 5.3 represents the hardware circuit at drowning condition

In above 5.3 circuit, both conductivity plates drowned in the water, so we can infer that the child was drowned and relay and oxygen pump gets activated. Then LCD displays the belt id of the child who got drowned. then buzzer raise the alarm. Finally SMS will sent to the guardian.

5.4 CIRCUIT DIAGRAM

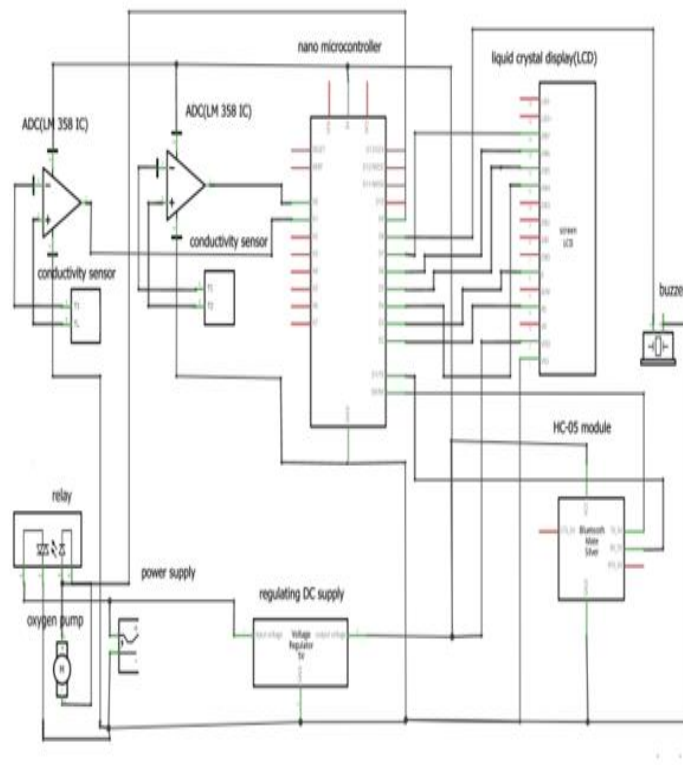


Fig 5.4 represents the Circuit Diagram

- The Nano microcontroller is the main part of the circuit. it controls the whole circuit.
- The power supply from the power source is given to regulator for converting 12v to 5v voltage.
- In Liquid crystal display the data pins are connected to Nano controller digital pin d2, d3, d4, d5, d6, d7 respectively. VSS pin is connected to ground and VDD pin is connected to 5v supply.

- ADC is used for converting analog signal to digital signal from the conductivity sensor. The output pins of ADC are connecting to controller analog pins.
- HC-05 wireless module has 4 pins. VCC pin is connecting to 5v, GND pin is connecting to ground. TX and RX pins are connecting to controller TX and RX pins.
- Relay is connecting to controller digital for activating the oxygen pump.
- Buzzer has two pins positive and negative. Positive pin is connecting to controller digital pin D8. negative pin is grounded.

Hardware Components

- Nano microcontroller
- Power supply
- ADC
- Conductivity sensors
- Relay
- Oxygen pump
- Buzzer
- LCD display
- HC-05 device

5.5 NANOMICROCONTROLLER

Here are few of its basic features which you must know if you are thinking to work on this great microcontroller board:

It has 22 input/output pins in total.

-
- 18mm
- MINI USB
- MEGA 328P
- D13
- 3V3
- REF
- A0
- A1
- A2
- A3
- A4
- A5
- A6
- A7
- 5V
- RST
- GND
- VIN
- D12
- D11
- D10
- D9
- D8
- D7
- D6
- D5
- D4
- D3
- D2
- GND
- RST
- RXD
- TX1
- 45mm
- Reset Button
- ICSP headers
- LED indicator

and finally, regulating to obtain a desired fixed DC voltage. The regulation is usually obtained from an IC voltage regulator unit, which takes a DC voltage and provides a somewhat lower DC voltage, which remains the same even if the input DC voltage varies, or the output load connected to the DC voltage changes.

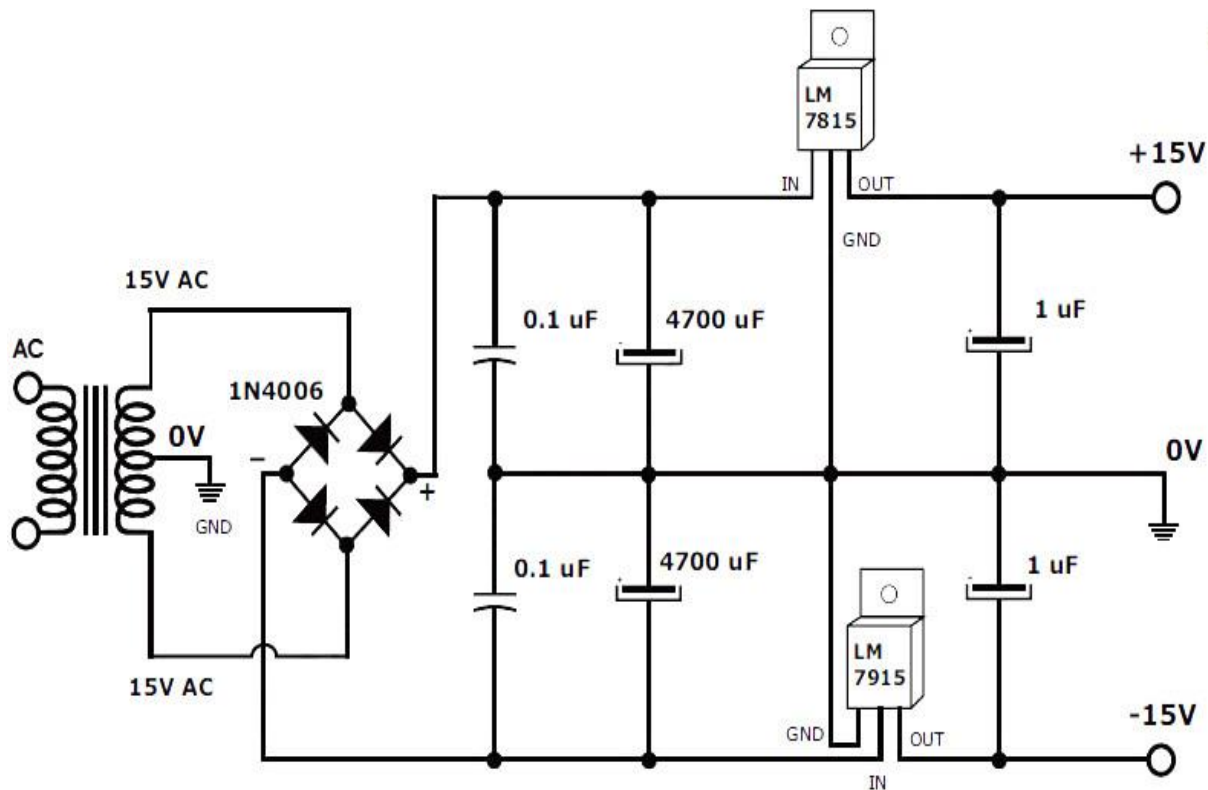


Fig 5.6 Power Supply Circuit Diagram

5.6.1 IC Voltage Regulator

Voltage regulators comprise a class of widely used ICs. Regulator IC units contain the circuitry for reference source, comparator amplifier, control device, and overload protection all in a single IC. Although the internal construction of the IC is somewhat different from that described for discrete voltage regulator circuits, the

external operation is much the same. IC units provide regulation of either a fixed positive voltage, a fixed negative voltage, or an adjustable set voltage.

A power supply can be built using a transformer connected to the AC supply line to step the AC voltage to desired amplitude, then rectifying that AC voltage, filtering with a capacitor and RC filter, if desired, and finally regulating the DC voltage using an IC regulator. The regulators can be selected for operation with load currents from hundreds of Milli amperes to tens of amperes, corresponding to power ratings from milli watts to tens of watts.

5.6.2 Three-Terminal Voltage Regulators

Fig shows the basic connection of a three-terminal voltage regulator IC to a load. The fixed voltage regulator has an unregulated DC input voltage, V_I , applied to one input terminal, a regulated output DC voltage, V_O , from a second terminal, with the third terminal connected to ground. For a selected regulator, IC device specifications list a voltage range over which the input voltage can vary to maintain a regulated output voltage over a range of load current. The specifications also list the amount of output voltage change resulting from a change in load current (load regulation) or in input voltage (line regulation).

5.7 ADC

An ADC converts a continuous-time and continuous-amplitude analog signal to a discrete-time and discrete-amplitude digital signal. The conversion involves division of the input, thus it essentially introduces a little quantity of error or noise. What is more, rather than endlessly playing the conversion, Associate in Nursing ADC will the conversion sporadically, sampling the input, limiting the allowable information measure of the signaling.

5.8 CONDUCTIVITY SENSOR

The device converts the measured insulator properties into Water Content over the complete vary, 0 80%, victimisationstandardisation tables. Generalized calibrations are provided for many common, and specialised calibrations are offered as separate price choices for variety of artificial substrates. The physical phenomenon device conjointly calculates Pore Water physical phenomenon, the Electrical physical phenomenon of the water among the pores (ECp). Its calculation is predicated on a singular formula that minimizes the results of probe contact and on the readings.

5.9 BLUETOOTH

- It is used for many applications like wireless headset, game controllers, wireless mouse, wireless keyboard and many more consumer applications.
- It has range up to <100m which depends upon transmitter and receiver, atmosphere, geographic & urban conditions.
- It is IEEE 802.15.1 standardized protocol, through which one can build wireless Personal Area Network (PAN). It uses frequency-hopping spread spectrum (FHSS) radio technology to send data over air.
- It uses serial communication to communicate with devices. It communicates with microcontroller using serial port (USART).

HC-05 Bluetooth Module

- HC-05 is a Bluetooth module which is designed for wireless communication. This module can be used in a master or slave configuration.

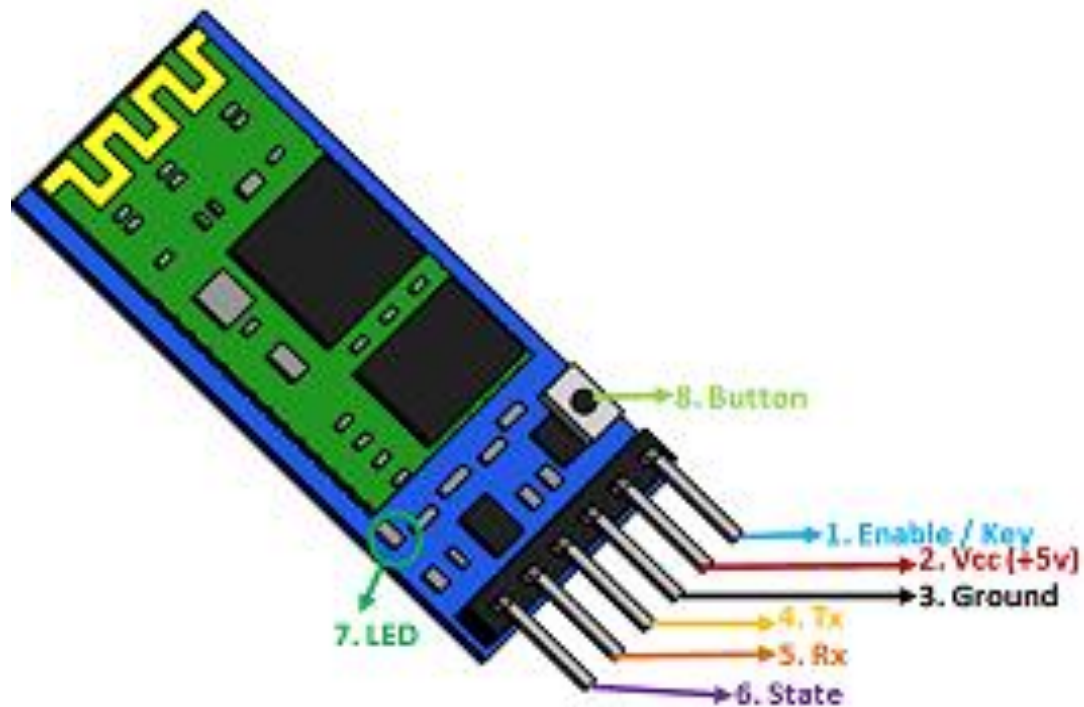


Fig 5.7 HC-05 Bluetooth

Bluetooth serial modules allow all serial enabled devices to communicate with each other using Bluetooth.

5.10 BUZZER

A buzzer could be a little nevertheless economical part to feature sound options to our project/system. it's terribly little and compact 2-pin structure thus is simply used on bread board, Perf Board and even on PCBs that makes this a wide used part in most electronic applications.

There are 2 sorts are buzzers that are unremarkably out there. The one shown here could be a easy buzzer that once power-driven can create a continual sound, the opposite kind is termed a readymade buzzer which can look bulkier than this and can manufacture a Beep. Beep. Beep. Sound because of the interior periodical circuit gift within it. But, the one shown here is most generally used as a result of it is customized with facilitate of alternative circuits to suit simply in our application.

This buzzer is utilized by merely powering it employing a DC power provide starting from 4V to 9V. an easy 9V battery also can be used, however it's suggested to use a regulated +5V or +6V DC provide.

5.11 LCD DISPLAY

Stands for "Liquid Crystal Display" {digital display|alphanumeric display}.it could be a flat panel display technology normally utilized in TVs and laptop monitors. it's additionally utilized in screens for mobile devices, like laptops, tablets, Sand smartphones. LCD displays do not simply look completely different than large cathode-ray tube monitors, the manner they operate is considerably completely different similarly. rather than firing electrons at a glass screen, associate digital display has backlight that gives lightweight to individual pixels organized during a rectangular grid. every element incorporates a red, green, and blue RGB sub-pixel which will be turned on or off. once all of a pixel's sub-pixels square measure turned off, it seems black. once all the sub-pixels square measure turned on 100%, it seems white. By adjusting the individual levels of red, green, and blue lightweight, legion color combos square measure attainable.

5.12 RELAY

An iron core is enclosed by an effect coil. As shown, the ability supply is given to the magnet through an effect switch and thru contacts to the load. once current starts flowing through the management coil, the magnet starts energizing and therefore intensifies the field. Thus, the higher tangency starts to be drawn to the lower fastened arm and therefore closes the contacts inflicting a brief circuit for the ability to the load. On the opposite hand,ifthe relay was already de-energized once the contacts were closed, then the contact move

oppositely associated degree build an circuit.

5.13 OXYGEN PUMP

An element pump may be a pump for pushing air. Examples embody a pump, pumps that square measure accustomed aerate Associate in Nursing marine museum or a lake via Associate in Nursing air stone; a gas mechanical device accustomed power a gas tool, air horn or pipe organ; a bellows accustomed encourage a fire; a home appliance and a pump. All element pumps contain a neighborhood that moves (vane, piston, impeller, diaphragm etc.) that drives the flow of air. once the air gets moved, a part of air mass gets created that fills up with additional air. Pumps and compressors use terribly similar mechanisms, and essentially perform a similar action, however in several fluid regimes. At some purpose there's a crossover purpose in nomenclature, however here square measure some stereotypes. Compressors care for compressible fluids, generally gases. Pumps care for fluids, generally liquids, approximated as in-compressible.



Fig 5.8 Oxygen Pump

5.14 ANDROID APPLICATION

ECLIPSE SOFTWARE

Eclipse started as a proprietary IBM product (IBM Visual age for Smalltalk/Java) – Embracing the open source model IBM opened the product up

- Open Source – It is a general purpose open platform that facilitates and encourages the development of third party plug-ins
- Best known as an Integrated Development Environment (IDE) – Provides tools for coding, building, running and debugging applications
- Originally designed for Java, now supports many other languages – Good support for C, C++ – Python, PHP, Ruby, etc...

Eclipse is written in Java and will thus need an installed JRE or JDK in which to execute – JDK recommended

5.15 Java

Android applications are developed using the Java language. As of now, that's really your only option for native applications. Java is a very popular programming language developed by Sun Microsystems (now owned by Oracle). Developed long after C and C++, Java incorporates many of the powerful features of those powerful languages while addressing some of their drawbacks. Still, programming languages are only as powerful as their libraries. These libraries exist to help developers build applications.

Some of the Java's important core features are:

- It's easy to learn and understand
- It's designed to be platform-independent and secure, using virtual machines

- It's object-oriented

5.16 BLOCK DIAGRAM OF ANDROID SOFTWARE

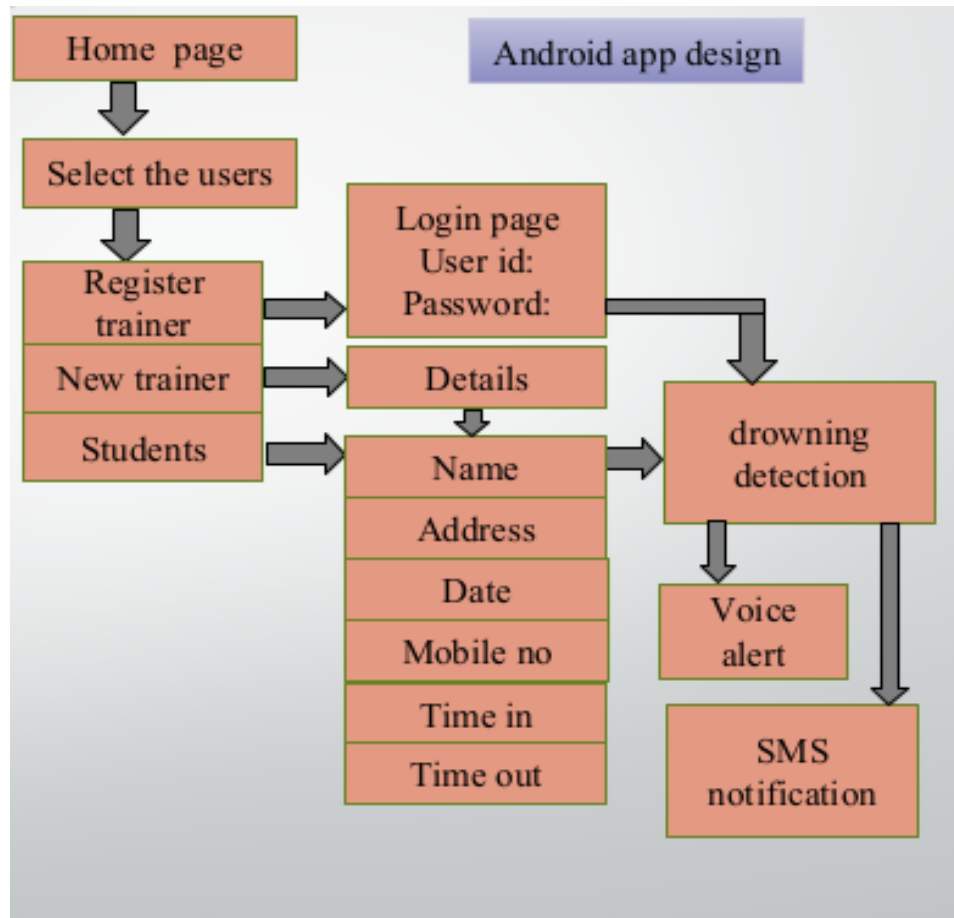


Fig 5.9 represents the block diagram of android software

This app home page shows like select the user mode. then it will ask whether you are a Register trainer or New trainer or Students. If we click the Register trainer means then it will ask user id and password of the child, then that child is connected to the hardware system. If you click New trainer means then it will ask the child to enter the details like Name ,Address,Date ,mobile number,time in,time out.If the

child clickas a student, then it will ask same details as New trainer. In this way the app is working.

5.17 PROCESS FLOWCHART FOR ANDROID APP

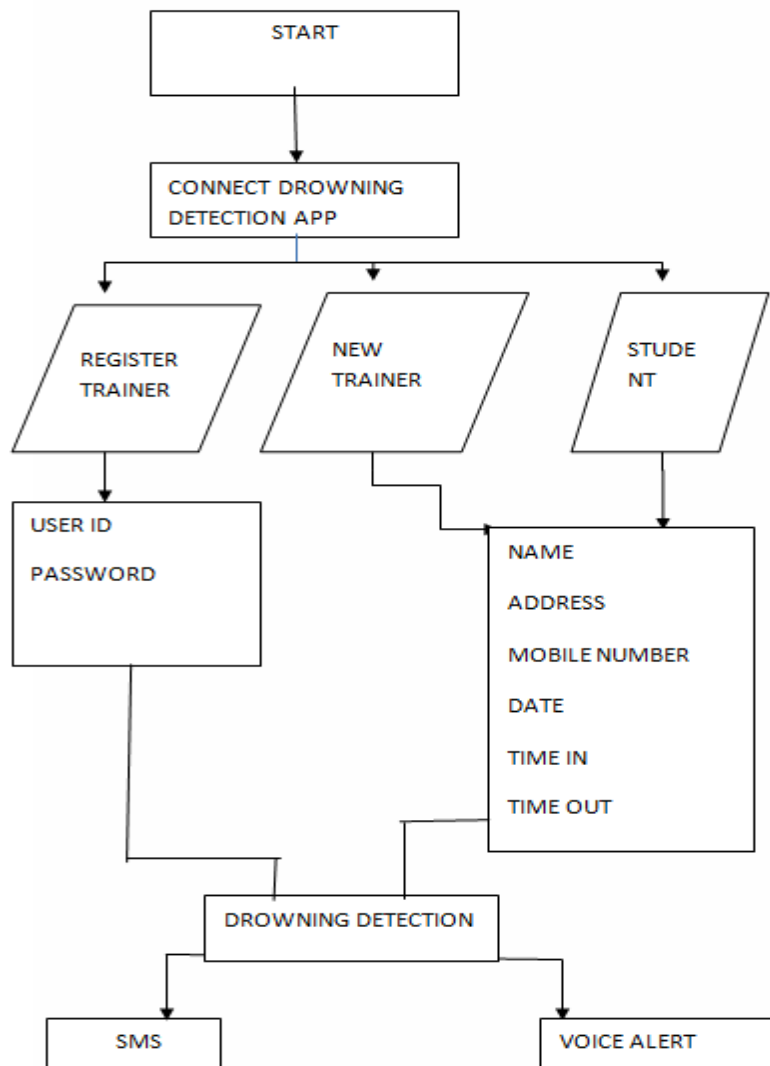


Fig 5.10 Process Flowchart

Fig 5.10 represents the process flowchart for android app

5.18 ANDROID SOFTWARE RESULTS

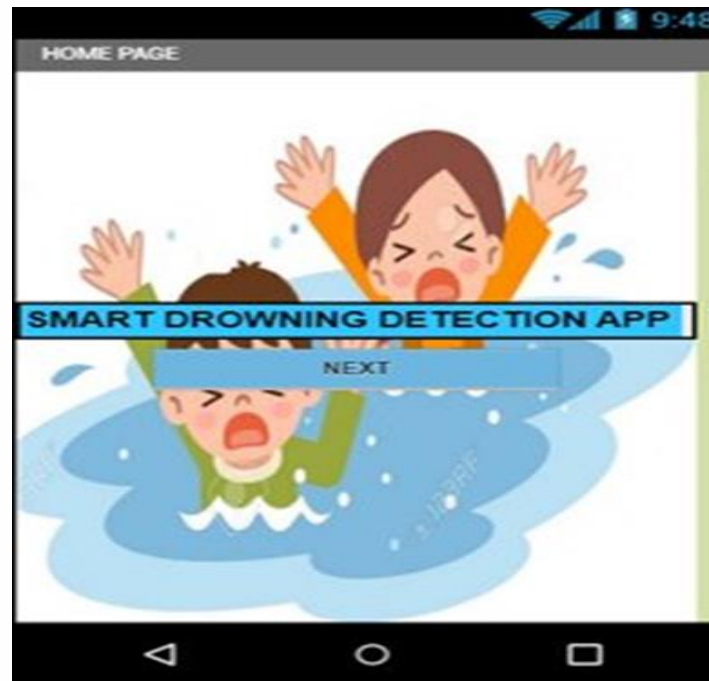


Fig 5.11 Represents the home page of the application



Fig 5.12 represents the select user mode page



DETAILS

NEW TRAINER DETAILS

NAME:

ADDRESS:

MOBILE NO:

DATE:

TIME IN:

TIME OUT:

SAVE

NEXT

This screenshot shows a mobile application interface for entering new trainer details. The title bar at the top is labeled 'DETAILS'. The main heading is 'NEW TRAINER DETAILS'. Below this, there is a form with six input fields: 'NAME:', 'ADDRESS:', 'MOBILE NO:', 'DATE:', 'TIME IN:', and 'TIME OUT:'. Each field is represented by a white rectangular box. Below the form, there are two blue buttons: 'SAVE' and 'NEXT'. The background of the form area features a cartoon character of a person with orange hair and a blue shirt. The bottom of the screen shows the standard Android navigation bar with back, home, and recent apps icons.

Fig 5.13 represents the new trainer details



LOGIN PAGE

LOGIN PAGE FOR REG TRAINER

ID NO:

PASSWORD:

SAVE

LOGIN

This screenshot shows a mobile application interface for the login page of a registered trainer. The title bar at the top is labeled 'LOGIN PAGE'. The main heading is 'LOGIN PAGE FOR REG TRAINER'. Below this, there is a form with two input fields: 'ID NO:' and 'PASSWORD:'. The 'ID NO:' field is a white rectangular box, and the 'PASSWORD:' field is a white rectangular box with a black border and a series of dots for masking. Below the form, there are two blue buttons: 'SAVE' and 'LOGIN'. The background of the form area features a cartoon character of a person in a white shirt and blue pants, standing in water and holding a lifebuoy. The bottom of the screen shows the standard Android navigation bar with back, home, and recent apps icons.

Fig 5.14 represents the login page for register trainer



Fig 5.15 represents the student details



Fig 5.16 represents the drowning detection page.

Fig 5.11-Fig 5.16 explains about android software results. Android app name is Smart drowning detection app. first, user need to open the application, then the user connect to Bluetooth connection to send the Sms to parents/guardians when user is under critical situation. Next, after connecting Bluetooth, it will show 3 modes like REG TRAINER,NEW TRAINER and STUDENTS. If the user clicks NEW TRAINER ,then it will ask the user to enter Name Address,Date ,Mobile number, Time in, Time out. After giving all these details, app will be connected. If the user clicks TRAINER, then it will ask the user to enter ID.NO,Password. Then, app will be connected. If the user clicks STUDENT, then it will ask the user to enter Name Address ,Date ,Mobile number, Time in, Time out. After giving all these details app will be connected.If the user drowning is detected,then his/her Belt id will be displayed immediately sms will be sent to the registered mobile number finally app will be disconnected.

CHAPTER 6

CONCLUSION

6.1 CONCLUSION

Consistently various individuals, together with youngsters, are suffocated or close to breathless within the deeps of the swimming pools, and also the lifeguards aren't ready all around to contend with these problems. during this manner raises the wants for having a framework which will so acknowledge the breathless individuals and alerted in period with voice alert and SMS notification, once the fearfulness. It will be put in in International standardized faculties wherever categories ar command for coaching youngsters.

6.2 FUTURE SCOPE

We can develop the algorithm for artificial intelligence. Through this ,we can monitor the child movement in real time by fixing the webcam in swimming pool. webcamtakes the picture of the child movement and match with database pictures,so we can get the child movements accurately.

REFERENCES

- [1] Alvin H. Kam, Wenmiao Lu and Wei-Yun Yau, "A Video-Based Drowning Detection System", Proceedings of the 7th European Conference on Computer Vision-Part IV, Copenhagen, Denmark, May 2018, pp. 297-311
- [2] Boulton, R. J. Micheals and X. Gao, "Into the Woods: Visual surveillance of noncooperative and camouflaged targets in complex outdoor settings," Proc. IEEE, vol. 89, no. 10, pp. 1382-1402, 2018.
- [3] Canny, "A computational approach to edge detection," IEEE Trans. Pattern Anal. Machine Intell., vol. 8, pp. 679-698, 2017.
- [4] Chi Zhang and Xiaoguang Li, "A Novel Camera-Based Drowning Detection Algorithm", Proceedings of Advances in Image and Graphics Technologies, Beijing, China, June 2018, pp. 224-23.
- [5] Duda, P. E. Hart and D. G. Stork, Pattern classification, John Wiley & Sons, 2020.
- [6] Friedman and S. Russell, "Image segmentation in video sequence: A probabilistic approach," Proc. Thirteenth Conf. Uncertainty in Artificial Intelligence, pp. 175-181, 2018.
- [7] Guichard, J.M. Lavest, B. Liege, J. Meniere, "Poseidon Technologies: The world's first and only computer-aided drowning detection system," Proc. IEEE Conf. Comp. Vision Patt. Recog., DT-8 (Demo Session), 2018.
- [8] Haritaoglu, D. Harwood, and L. S. Davis, "W4: Real-time surveillance of people and their activities," IEEE Trans. Pattern Anal. Machine Intell., 22, pp. 809-830, 2020.
- [9] Intille and J. Davis, "Real-time closed-world tracking" Proc. IEEE Conf. Comp. Vision Patt. Recog., pp. 697-703, 2018.

- [10] Kam, W. Lu and W.-Y. Yau, "A video-based drowning detection system," Springer-VerlagLec. Notes. Comp. Sci. vol. 2353 (ECCV 2002), Part IV, pp. 297-311, 2017.
 - [11] Lei Fei, Wang Xueli and Chen Dongsheng, "Drowning Detection Based on Background Subtraction", International Conferences on Embedded Software and System, 2017, pp. 341-343
 - [12] Meniere, "System for monitoring a swimming pool to prevent drowning accidents," US. Pat. No. 6,133,838, October 2019.
 - [13] Menoud, "Alarm and monitoring device for the presumption of bodies in danger in a swimming pool," US. Pat. No. 5,886,630, March 2017.
 - [14] Pavlidis, V. Morellas and P. Tsiamyrtzis and S. Harp, "Urban surveillance systems: From the laboratory to the commercial world," Proc. IEEE, vol. 89, no. 10, pp. 1478-1497, 2018.
 - [15] Pia, "The RID factor as a cause of drowning," Parks and Recreation, 2016.
 - [16] Toh and W.-Y. Yau, "Multi-modal biometrics fusion: Beyond optimal weighting," Proc. the Seventh Intl. Conf. Automation, Robotics and Vision, pp. 788-792,
 - [17] Wai Kit Wong, Joe How Hui, Chu Kiong Loo and Way Soong Lim, "Off-time swimming pool surveillance using thermal imaging system", International journal of innovative computing, information and control, vol. 9 (3), 2017, pp. 366-371
 - [18] Wren, A. Azarbayehani, T. Darrell, T. and A. Pentland, "Pfinder: Real-time tracking of the human body," IEEE Trans. Pattern Anal. Machine Intell.,2019
- data, respectively.
- [19] Won, "A block-based MAP segmentation for image compression," IEEE Trans. Circuits System Video Technology, vol. 8, no. 5, pp. 592-601,2020.

[20] Zou Xu; Wang Tingjun; Liu Lujun; Liao Zhonghao; Fan Jiayang; Zhang Yuanfei; Zeng Shun, Swimming Pool Anti-Drowning Monitoring System. CN107134116A 2017.

APPENDIX SOURCE CODE
PROGRAM FOR KEIL SOFTWARE

```
#define F_CPU 16000000UL          //define CPU frequency
#include<avr/io.h>
#include<util/delay.h>
#include<avr/interrupt.h>
#include "avr/io.h"
int condutivity_sensor_1 =9;
int condutivity_sensor_2 =10;
int buzzer = 8;
const int _pin = A4;
{
  lcd.setCursor(0,0);
  lcd.print("Drowning detect");
  lcd.setCursor(0,1);
  lcd.print("auto detect");
  delay(2500);
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("send to trainner");
  delay(2500);
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("DETECTING..");
  delay(2500);
```

```

lcd.clear();
lcd.setCursor(0,0);
lcd.print("connecting..");
    delay(2500);
lcd.clear();
lcd.setCursor(0,0);
lcd.print("sensor calibrating");
    delay(2500);
lcd.clear();
pinMode(wet_sensor1,INPUT);
pinMode(wet_sensor2,INPUT);
pinMode(buzzer,OUTPUT);
digitalWrite(buzzer,LOW);
digitalWrite(relay,LOW);
}
void LCD_Command(unsigned char cmnd)
{
    LCD_Data_Port= cmnd;
    LCD_Command_Port&= ~(1<<RS); /* RS=0 command reg. */
    LCD_Command_Port&= ~(1<<RW); /* RW=0 Write operation */
    LCD_Command_Port |= (1<<EN); /* Enable pulse */
    _delay_us(1);
    LCD_Command_Port&= ~(1<<EN);
    _delay_ms(3);
}
void LCD_Char (unsigned char char_data) /* LCD data write function */
{

```

```

    LCD_Data_Port = char_data;
    LCD_Command_Port |= (1<<RS);    /* RS=1 Data reg. */
    LCD_Command_Port&= ~(1<<RW); /* RW=0 write operation */
    LCD_Command_Port |= (1<<EN);    /* Enable Pulse */
    _delay_us(1);
    LCD_Command_Port&= ~(1<<EN);
    _delay_ms(1);
}

void LCD_String (char *str)
{
    inti;
    for(i=0;str[i]!=0;i++) /* send each char of string till the NULL */
    {
        LCD_Char (str[i]); /* call LCD data write */
    }
}

void LCD_String_xy (char row, char pos, char *str) {
    if (row == 0 &&pos<16)
        LCD_Command((pos& 0x0F)|0x80);
    else if (row == 1 &&pos<16)
        LCD_Command((pos& 0x0F)|0xC0);
    LCD_String(str);
}

void LCD_Clear(){
    LCD_Command (0x01);    /* clear display */
    LCD_Command (0x80);    /* cursor at home position */
}

```

```

    void loop()
    {
        if(digitalRead(touch)==HIGH)
        {
            lcd.setCursor(0,1);
            lcd.print(" CLICKED");
            digitalWrite(buzzer,HIGH);
            delay(500);
            Serial.println("ATEO");
            delay(500);
            Serial.println("AT+CMGF=1");
            delay(500);
            Serial.print("AT+CMGS=");
            delay(500);
            Serial.write(0x22);
            Serial.write(0x22);
            Serial.write(0x0D);
            Serial.write(0x0A);
            delay(500);
            Serial.println("S");
            delay(500);
            Serial.write(0x1a);
            delay(3000);
        }
    }

    void loop() {
        lcd.setCursor(0,0);
        lcd.print("Drowning monitor");
    }

```

```

int wet_calibrate1=analogRead(wet_sensor1);
int wet_calibrate2=analogRead(wet_sensor2);
Serial.print(wet_calibrate);
    if(wet_calibrate1>550) {
        Serial.print(wet_calibrate1);
        lcd.setCursor(0,1);
        lcd.print("drowning found 1");
        digitalWrite(buzzer,HIGH);
        digitalWrite(relay,HIGH);
        Serial.print("d");
        delay(1000);
        lcd.clear();    }
        if(wet_calibrate2>550)
            {
                Serial.print(wet_calibrate2);
                lcd.setCursor(0,1);
                lcd.print("drowning found 2");
                digitalWrite(buzzer,HIGH);
                digitalWrite(relay,HIGH);
                Serial.print("r");
                delay(1000);
                lcd.clear();    } else
            {
                lcd.setCursor(0,1);
                lcd.print("normal condition");
                digitalWrite(buzzer,LOW);
                digitalWrite(relay,LOW);    }}

```

PROGRAM FOR ANDROID SOFTWARE

```
package android.support.p000v4.app;
import android.app.Notification;
import android.os.Binder;
import android.os.IBinder;
import android.os.IInterface;
import android.os.Parcel;
import android.os.RemoteException;
/* renamed from: android.support.v4.app.INotificationSideChannel */
public interface INotificationSideChannel extends IInterface {
    void cancel(String str, inti, String str2) throws RemoteException;
    void cancelAll(String str) throws RemoteException;
    void notify(String str, inti, String str2, Notification notification) throws
RemoteException;
    /* renamed from: android.support.v4.app.INotificationSideChannel$Stub */
    public static abstract class Stub extends Binder implements
INotificationSideChannel {
        private static final String DESCRIPTOR =
"android.support.v4.app.INotificationSideChannel";
        static final int TRANSACTION_cancel = 2;
        static final int TRANSACTION_cancelAll = 3;
        static final int TRANSACTION_notify = 1;
        public Stub() {
attachInterface(this, DESCRIPTOR);
        }
        public static INotificationSideChannelasInterface(IBinderobj) {
```

```

        if (obj == null) {
            return null;
        }
        IInterfaceiin = obj.queryLocalInterface(DESCRIPTOR);
        if (iin == null || !(iin instanceof INotificationSideChannel)) {
            return new Proxy(obj);
        }
        return (INotificationSideChannel) iin;
    }

    public IBinder asBinder() {
        return this;
    }

    public boolean onTransact(int code, Parcel data, Parcel reply, int flags) throws
RemoteException {
        Notification _arg3;
        switch (code) {
            case 1:
data.enforceInterface(DESCRIPTOR);
                String _arg0 = data.readString();
int _arg1 = data.readInt();
                String _arg2 = data.readString();
                if (data.readInt() != 0) {
                    _arg3 = (Notification)
Notification.CREATOR.createFromParcel(data);
                } else {
                    _arg3 = null;
                }

```

```

        notify(_arg0, _arg1, _arg2, _arg3);
        return true;
    case 2:
data.enforceInterface(DESCRIPTOR);
        cancel(data.readString(), data.readInt(), data.readString());
        return true;
    case 3:
data.enforceInterface(DESCRIPTOR);
cancelAll(data.readString());
        return true;
    case 1598968902:
reply.writeString(DESCRIPTOR);
        return true;
    default:
        return super.onTransact(code, data, reply, flags);
    }
}

private static class Proxy implements INotificationSideChannel {
    private IBinderRemote;
    Proxy(IBinder remote) {
this.mRemote = remote;
    }
    public IBinder asBinder() {
        return this.mRemote;
    }
    public String getInterfaceDescriptor() {
        return Stub.DESSCRIPTOR;
    }

```

```

    }

    public void notify(String packageName, int id, String tag, Notification
notification) throws RemoteException {
        Parcel _data = Parcel.obtain();
        try {
            _data.writeInterfaceToken(Stub.DESCRITOR);
            _data.writeString(packageName);
            _data.writeInt(id);
            _data.writeString(tag);
            if (notification != null) {
                _data.writeInt(1);
notification.writeToParcel(_data, 0);
            } else {
                _data.writeInt(0);
            }
this.mRemote.transact(1, _data, (Parcel) null, 1);
        } finally {
            _data.recycle();
        }
    }

    public void cancel(String packageName, int id, String tag) throws
RemoteException {
        Parcel _data = Parcel.obtain();
        try {
            _data.writeInterfaceToken(Stub.DESCRITOR);
            _data.writeString(packageName);
            _data.writeInt(id);

```

```

        _data.writeString(tag);
this.mRemote.transact(2, _data, (Parcel) null, 1);
    } finally {
        _data.recycle();
    }
}

public void cancelAll(String packageName) throws RemoteException {
    Parcel _data = Parcel.obtain();
    try {
        _data.writeInterfaceToken(Stub.DESRIPTOR);
        _data.writeString(packageName);
this.mRemote.transact(3, _data, (Parcel) null, 1);
    } finally {
        _data.recycle();
    }
}
}

package android.support.p000v4.graphics.drawable;
import androidx.annotation.RestrictTo;
import androidx.core.graphics.drawable.IconCompat;
import androidx.versionedparcelable.VersionedParcel;
@RestrictTo({RestrictTo.Scope.LIBRARY})
public final class IconCompatParcelizer extends
androidx.core.graphics.drawable.IconCompatParcelizer {
    public static IconCompat read(VersionedParcel parcel) {

```

```

        return androidx.core.graphics.drawable.IconCompatParcelizer.read(parcel);
    }

    public static void write(IconCompatobj, VersionedParcel parcel) {
        androidx.core.graphics.drawable.IconCompatParcelizer.write(obj, parcel);
    }
}

package android.support.p000v4.p001os;
import android.os.Binder;
import android.os.Bundle;
import android.os.IBinder;
import android.os.IInterface;
import android.os.Parcel;
import android.os.RemoteException;
public interface IResultReceiver extends IInterface {
    void send(inti, Bundle bundle) throws RemoteException;
    public static abstract class Stub extends Binder implements IResultReceiver {
        private static final String DESCRIPTOR =
"android.support.v4.os.IResultReceiver";
        static final int TRANSACTION_send = 1;
        public Stub() {
attachInterface(this, DESCRIPTOR);
        }
        public static IResultReceiverasInterface(IBinderobj) {
            if (obj == null) {
                return null;
            }
IInterfaceiin = obj.queryLocalInterface(DESCRIPTOR);

```

```

        if (iin == null || !(iin instanceof IResultReceiver)) {
            return new Proxy(obj);
        }
        return (IResultReceiver) iin;
    }

    public IBinder asBinder() {
        return this;
    }

    public boolean onTransact(int code, Parcel data, Parcel reply, int flags) throws
RemoteException {
        Bundle _arg1;
        switch (code) {
            case 1:
data.enforceInterface(DESCRIPTOR);
int _arg0 = data.readInt();
                if (data.readInt() != 0) {
                    _arg1 = (Bundle) Bundle.CREATOR.createFromParcel(data);
                } else {
                    _arg1 = null;
                }
                send(_arg0, _arg1);
                return true;
            case 1598968902:
reply.writeString(DESCRIPTOR);
                return true;
            default:
                return super.onTransact(code, data, reply, flags);
        }
    }

```

```

    }
}
private static class Proxy implements IResultReceiver {
    private IBinder mRemote;
    Proxy(IBinder remote) {
this.mRemote = remote;
    }
    public IBinder asBinder() {
        return this.mRemote;
    }
    public String getInterfaceDescriptor() {
        return Stub.DESRIPTOR;
    }
    public void send(int resultCode, Bundle resultData) throws
RemoteException {
        Parcel _data = Parcel.obtain();
        try {
            _data.writeInterfaceToken(Stub.DESRIPTOR);
            _data.writeInt(resultCode);
            if (resultData != null) {
                _data.writeInt(1);
resultData.writeToParcel(_data, 0);
            } else {
                _data.writeInt(0);
            }
this.mRemote.transact(1, _data, (Parcel) null, 1);
        } finally {

```

```

        _data.recycle();        }    }    }    }
    }

package android.support.p000v4.p001os;
import android.os.Bundle;
import android.os.Handler;
import android.os.Parcel;
import android.os.Parcelable;
import android.os.RemoteException;
import android.support.p000v4.p001os.IResultReceiver;
import androidx.annotation.RestrictTo;
@RestrictTo({RestrictTo.Scope.LIBRARY_GROUP})
/* renamed from: android.support.v4.os.ResultReceiver */
public class ResultReceiver implements Parcelable {
    public static final Parcelable.Creator<ResultReceiver> CREATOR = new
Parcelable.Creator<ResultReceiver>() {
        public ResultReceivercreateFromParcel(Parcel in) {
            return new ResultReceiver(in);
        }

        public ResultReceiver[] newArray(int size) {
            return new ResultReceiver[size];
        }
    };
    final Handler mHandler;
    final boolean mLocal;
    IResultReceiver mReceiver;
    /* renamed from: android.support.v4.os.ResultReceiver$MyRunnable */

```

```

class MyRunnable implements Runnable {
    final int mResultCode;
    final Bundle mResultData;
    MyRunnable(int resultCode, Bundle resultData) {
        this.mResultCode = resultCode;
        this.mResultData = resultData;
    }
    public void run() {
        ResultReceiver.this.onReceiveResult(this.mResultCode, this.mResultData);
    }
}

class MyResultReceiver extends IResultReceiver.Stub {
    MyResultReceiver() {
    }
    public void send(int resultCode, Bundle resultData) {
        if (ResultReceiver.this.mHandler != null) {
            ResultReceiver.this.mHandler.post(new MyRunnable(resultCode, resultData));
        } else {
            ResultReceiver.this.onReceiveResult(resultCode, resultData);
        }
    }
    public ResultReceiver(Handler handler) {
        this.mLocal = true;
        this.mHandler = handler;
    }
    public void send(int resultCode, Bundle resultData) {
        if (this.mLocal) {
            if (this.mHandler != null) {
                this.mHandler.post(new MyRunnable(resultCode, resultData));
            }
        }
    }
}

```

```

        } else {
onReceiveResult(resultCode, resultData);
        }
        } else if (this.mReceiver != null) {
            try {
this.mReceiver.send(resultCode, resultData);
            } catch (RemoteException e) {
            }
        }
    public void onReceiveResult(intresultCode, Bundle resultData) {
    }
    public intdescribeContents() {
        return 0;
    }
    public void writeToParcel(Parcel out, int flags) {
        synchronized (this) {
            if (this.mReceiver == null) {
this.mReceiver = new MyResultReceiver();
            }
            out.writeStrongBinder(this.mReceiver.asBinder());
        }
    }
    ResultReceiver(Parcel in) {
this.mLocal = false;
this.mHandler = null;
this.mReceiver = IResultReceiver.Stub.asInterface(in.readStrongBinder());
    }

```