ASSESSMENT OF WATER LEVEL INDICATOR USING AURDINO AND ULTRASONIC SENSOR LCD DISPLAY

MD IMRAN HOSSAIN

SHAHRIAR RASHID (JYAMI)

MD. AL-AMIN

MONIRUZZAMAN

MD SOWKAT



DEPARTMENT OF MECHANICAL ENGINEERING

SONARGAON UNIVERSITY

MAY-2025

ASSESSMENT OF WATER LEVEL INDICATOR USING AURDINO AND ULTRASONIC SENSOR LCD DISPLAY

A Thesis Submitted By

Name Of The Student	**Student ID**
MD IMRAN HOSSAIN	: ME2103025015
SHAHRIAR RASHID (JYAMI)	: ME2101023465
MD. AL-AMIN	: ME2103025123
MONIRUZZAMAN	: ME2103025251
MD SOWKAT	: ME2103025252



Supervisor:

M. I. Washif Rahman

Lecturer

Submitted to the

DEPARTMENT OF MECHANICAL ENGINEERING

SONARGAON UNIVERSITY

May-2025

STUDENT DECLARATION

This is to certify that the thesis entitled ―Assessment of Water level indicator using Arduino ultrasonic sensor LCD display ” is an outcome of the investigation carried out by the author under the supervision of M.I. Washif Rahman Lecturer, Dept. of Mechanical Engineering, Sonargaon University (SU). This thesis or any part of it has not been submitted elsewhere for the award of any other degree or diploma or other similar title or prize.

SUBMITTED BY

MD IMRAN HOSSAIN
ME2103025015

SHAHRIAR RASHID (JYAMI) ID:
ID: ME2101023465

MD. AL-AMIN
ME2103025123

MONIRUZZAMAN ID:
ID: ME2103025251

MD SOWKAT
ID: ME2103025252

APPROVED BY

M. I. Washif Rahman
Lecturer
Department of Mechanical Engineering

Sonargaon University (SU)
Dhaka, Bangladesh

ABSTRACT

The Water Level Indicator system is designed to monitor and display the water level in a tank or container using an Arduino microcontroller and an ultrasonic sensor. This project aims to automate the process of water level detection, reducing the need for manual checks and ensuring efficient water usage. The ultrasonic sensor emits sound waves that reflect off the water surface, and the time it takes for the echo to return is used to calculate the distance between the sensor and the water. The Arduino processes this information and calculates the water level accordingly. The water level is then displayed on an LCD screen, providing real-time updates to the user. A visual display helps in understanding the exact level of water, while a buzzer or other indicators can be used to alert users when the water reaches certain critical levels (e.g., full or low). This project is particularly useful in applications such as household water tanks, industrial systems, and agricultural setups, where timely monitoring and control of water levels are essential. This report describes the design and implementation of. a water level indicator system using an Arduino microcontroller, ultrasonic sensor (HC-SR04), and a relay-controlled water pump. The system monitors water levels in both the overhead and reserve tanks and automatically activates the pump to transfer water when necessary. The objective is to automate water management for improved efficiency and monitoring.

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to everyone who supported me throughout the development of this Arduino Water Level Indicator project. Firstly, I would like to thank my project guide, [Instructor's Name], for providing valuable guidance, constant encouragement, and expert advice. Their insights and feedback were instrumental in the successful completion of this project. I would also like to extend my gratitude to my family and friends for their continuous support, patience, and motivation during the course of this project. Their belief in my abilities helped me stay focused and committed. I would like to acknowledge the contributions of the Arduino community, whose open-source resources and forums provided me with a wealth of information that made the implementation of the system possible. The availability of readily accessible tutorials and documentation greatly enhanced my understanding of the tools and techniques required.

INDEX

SL.NO	TITLE	PAGE NO
Chapter-1	Introduction	1
	Objective	
	Apparatus	2
Chapter-2	Materials And Components	3
	Digital Pins	
	Arduino Uno Board	
	Power Pins	
	Applications	
	Software	
	Sketch	
	Power (Usb / Barrel Jack)	
	Reset Button	
	Main Ice	12
	Ultrasonic Sensor (Hc-Sr04)	13
	Figure	
	Applications	15
	Board Prototype Circuit Bus Breadboard	16
	Relay Module	17
	Relay	18
	Water Pump	19
	Pump Figure	20

	16x2 LCD Display Picture	21
	I2c – Inter Integrated Circuit	22
	LCD Features	23
	Jumper Wire	24
	Jumper Wire Features	25
	Power Supply	26
	2pin Switch Specifications	27
	Led Traffic Light Module 5v	28
	Water Level Control Switch	29
	Passive Buzzers Module	30
Chapter-3	Circuit Diagram	31
	Ultrasonic Sensor Connection	
	Measure the Distance	33
Chapter-4	Block Diagram	34
	System Design	
	Water Conservation	
	Power Dependency	37
Chapter-5	Calculation Table	38
	Graph	
	Result	39
Chapter-6	Conclusion	40

Chapter-7	Program Code	41
	Program Code	43
Chapter-8	Reference	44
	Recommendation	45
Chapter-9	Appendix	46

Chapter-1

Introduction

The facility requirements in many industries, farms, hostels, hotels, offices include an overhead tank for water, which is usually fed through an electric pump that is switched off when the tank is filled up and switched on when it is empty. so, the most common way of knowing when the tank is filled is by observing when it overflows the brim. depending on the type of liquid being handled, overfilling of such a tank could lead to a great liquid material losses ranging in the order of thousands of naira per week depending on the extent of such application. these losses can be prevented if the tank is monitored automatically by incorporating a feedback. water level indicator using ultrasonic sensor & arduino is an amazing and very useful project. the objective of this project is to notify the user the amount of water that is present in the overhead water tank. this project can be further enhanced to control the water level in the tank by turning it on, when the water level is low, and turning it off when the water level is high. thus, the arduino water level indicator helps in preventing wastage of water in overhead tank. a transmitter circuit and a receiver circuit. the transmitter circuit makes use of an ultrasonic sensor to measure the water level in terms of distance. this data is sent to the receiver circuit using if communication. Water management in overhead tanks is crucial for ensuring consistent water supply and preventing overflow or shortages. Traditional methods of monitoring water levels often require manual checks, which can be inefficient and error-prone. This project aims to automate this process using an Arduino-based system with an ultrasonic sensor to monitor water levels in real-time and control a pump to maintain optimal water levels.

Objective

The objective of the project is to design a water level monitoring system that:

1. Accurately measure water level using an ultrasonic sensor.
2. Display the water level in real-time on an LCD screen.
3. Present the information in a user-friendly format.
4. Implement low and high water level alerts.
5. Display the water level as a percentage of tank capacity.
6. Include a calibration feature.

Apparatus

1. Aurdino board Uno R3
2. Ultrasonic sensor
3. Capacitor
4. Jumper wires
5. LED (Red and Green)
6. Water tank (Overhead and Underground)
7. Bus board
8. Adaptor 12volt
9. Buzzer
10. LCD Display
11. Water Level Sensor Plastic PP Liquid Water Switch
12. Relay Module
13. Water Pump
14. Spst 2pin switch red mini push bottom

Chapter-2

Materials and components

Arduino Uno board

Procedure introduction to the arduino board looking at the board from the top down, this is an outline of what you will see (parts of the board you might interact with in the course of normal use are highlighted)

Starting clockwise from the top center

- analog reference pin (orange)
- digital ground (light green)
- digital pins 2-13 (green)
- digital pins 0-1/serial in/out - tx/rx (dark green) - these pins cannot be used for digital i/o (digital read and digital write) if you are also using serial communication (e.g. serial. begin).
- reset button - s1 (dark blue) xi
- in-circuit serial programmer (blue-green)
- analog in pins 0-5 (light blue)
- power and ground pins (power: orange, grounds: light orange)
- external power supply in (9-12vdc) - x1 (pink)
- toggles external power and usb power (place jumper on two pins closest to desired supply) - sv1 (purple)
- usb (used for uploading sketches to the board and for serial communication between the board and the computer; can be used to power the board) (yellow)

Digital pins

In addition to the specific functions listed below, the digital pins on an arduino board can be used for general purpose input and output via the pin mode(), digital read(),and digital write () comm-ands. each pin has an internal pull-up resistor which can be turned on and off using digital write() (w/ a value of high or low, respectively) when the pin is configured as an input. the maximum current per pin is 40 ma. .serial: 0 (rx) and 1 (tx). used to receive (rx) and transmit (tx) ttl serial data. on the arduino diecimila, these pins are connected to the corresponding pins of the ftdi usb-to-ttl serial chip. on the arduino bt, they are connected to the corresponding pins of the wt11 bluetooth module. on the arduino mini and lily pad arduino, they are intended for use with an external ttl serial module (e.g. the mini-usb adapter).

- external interrupts: 2 and 3. these pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. see the attach interrupt() function for details.
- pwm: 3, 5, 6, 9, 10, and 11. provide 8-bit pwm output with the analog write() function. on boards with an atmega8, pwm output is available only on pins 9, 10, and 11.
- bt reset: 7. (arduino bt-only) connected to the reset line of the bluetooth module.
- spi: 10 (ss), 11 (mosi), 12 (miso), 13 (sck). these pins support spi communication, which, although provided by the underlying hardware, is not currently included in the arduino language.
- led: 13. on the diecimila and lily pad, there is a built-in led connected to digital pin 13. when the pin is high value, the led is on, when the pin is low, it's off.

The microcontroller develop ment board that will be at the heart of your projects. It’s a simple computer, but one that has no way for you to interact with it yet. You will be building the circuits and interfaces for interaction, and telling the microcontroller how to interface with other components.

Arduino Uno board

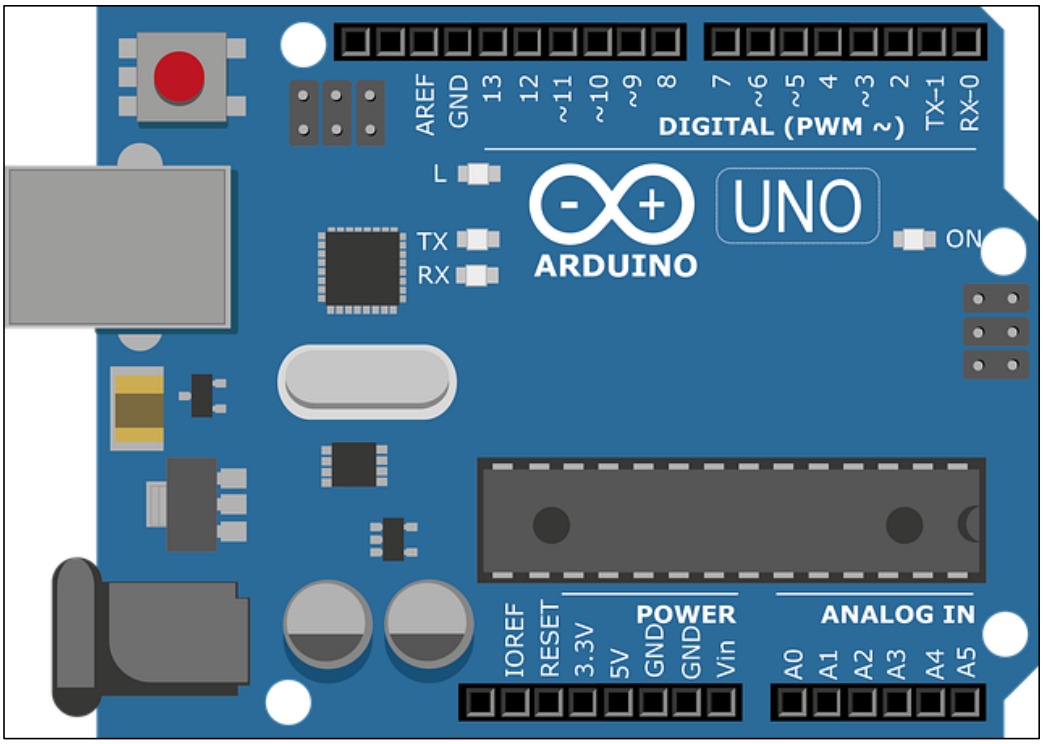


Figure no-1: Arduino Uno r3 [1]

Analog pins

In addition to the specific functions listed below, the analog input pins support 10-bit analog- to-digital conversion (adc) using the analog read () function. most of the analog inputs can also be used as digital pins: analog input 0 as digital pin 14 through analog input 5 as digital pin 19. analog inputs 6 and 7 (present on the mini and bt) cannot be used as digital pins.

- i2c: 4 (sda) and 5 (scl). support i2c (twi) communication using the wire library (documentation on the wiring website).

Power pins

- vin (sometimes labelled "9v"). the input voltage to the arduino board when it's using an external power source (as opposed to 5 volts from the usb connection or other regulated power source). you can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin. note that different boards accept different input voltages ranges, please see the documentation for your board. also note that the lily pad has no vin pin and accepts only a regulated input.
- 5v. the regulated power supply used to power the microcontroller and other components on the board. this can come either from vin via an on-board regulator, or be supplied by usb or another regulated 5v supply.
- 3v3. (diecimila-only) a 3.3 volt supply generated by the on-board ftdi chip.
- gnd. ground pins.

Other pins

- aref. reference voltage for the analog inputs. not currently supported by the arduino software.
- reset. (decimal-only) bring this line low to reset the microcontroller. typically used to add a reset button to shields which block the one on the board.

The text of the arduino getting started guide is licensed under a creative commons attribution- share alike 3.0 license. code samples in the guide are released into the public domain.

arduino is an open-source hardware and software company, project and user community that designs and manufactures single-board microcontrollers and microcontroller kits for building digital devices. its products are licensed under the gnu lesser general public license (lgpl) or the gnu general public license (gpl), permitting the manufacture of arduino boards and software distribution by anyone. arduino boards are available commercially in preassembled form or as do-it-yourself (diy) kits.

Arduino board designs use a variety of microprocessors and controllers. the boards are equipped with sets of digital and analog input/output (i/o) pins that may be interfaced to various expansion boards ('shields') or breadboards (for prototyping) and other circuits. the boards feature serial communications interfaces, including universal serial bus (usb) on some models, which are also used for loading programs from personal computers. the microcontrollers can be programmed using c and c++ programming languages. in addition to using traditional compiler tool chains, the arduino project provides an integrated development environment (ide) based on the processing language project.

the arduino project started in 2005 as a program for students at the interaction design institute ivrea in ivrea, italy, aiming to provide a low-cost and easy way for novices and professionals to create devices that interact with their environment using sensors and actuators. common examples of such devices intended for beginner hobbyists include simple robots, thermostats and motion detectors. the name arduino comes from a bar in ivrea, italy, where some of the founders of the project used to meet. the bar was named after arduino of ivrea, who was the margrave of the march of ivrea and king of Italy from 1002 to 1014

Applications

- Arduboy, a handheld game console based on arduino
- Arduino me, a midi controller device that mimics the monomer
- Ardupilot, drone software and hardware
- Ardusat, a cubesat based on arduino.
- C-stem studio, a platform for hands-on integrated learning of computing, science, technology, engineering, and mathematics (c-stem) with robotics.
- Data loggers for scientific research.
- Obduino, a trip computer that uses the on-board diagnostics interface found in most modern cars
- Openevse an open-source electric vehicle charger
- Xod, a visual programming language for arduino

Software

A program for arduino hardware may be written in any programming language with compilers that produce binary machine code for the target processor. Atmel provides a development environment for their 8-bit avr and 32-bit arm cortex-m based microcontrollers: avr studio (older) and atmel studio (newer).

Ide

The arduino integrated development environment (ide) is a cross-platform application (for windows, macos, and linux) that is written in the programming language java. it originated from the ide for the languages processing and wiring. it includes a code editor with features such as text cutting and pasting, searching and replacing text, automatic indenting, brace matching, and syntax highlighting, and provides simple one- click mechanisms to compile and upload programs to an arduino board. it also contains a message area, a text console, a toolbar with buttons for common functions and a hierarchy of operation menus. the source code for the ide is released under the gnu general public license, version 2. the arduino ide supports the languages c and c++ using special rules of code structuring. the arduino ide supplies a software library from the wiring project, which provides many common input and output procedures. user-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub main() into an executable cyclic executive program with the gnu tool chain, also included with the ide distribution. the arduino ide employs the program avrdude to convert the executable code into a text file in hexadecimal encoding that is loaded into the arduino board by a loader program in the board's firmware.

Pro ide

On October 18th, 2019, arduino pro ide (alpha preview) was released. the system still uses arduino cli (command line interface), but improvements include a more professional

development environment, auto completion support, and git integration. the application frontend is based on the eclipse their open source ide. the main features available in the alpha release are:

- modern, fully featured development environment
- dual mode, classic mode (identical to the classic arduino ide) and pro mode (file system view)
- new board manager
- new library manager
- board list
- basic auto-completion (arm targets only)
- git integration
- serial monitor
- dark mode

Sketch

A sketch is a program written with the arduino ide sketches are saved on the development computer as text files with the file extension . ino. arduino software (ide) pre-1.0 saved sketches with the extension .pde. a minimal arduino c/c++ program consists of only two functions

- Setup : this function is called once when a sketch starts after power-up or reset. it is used to initialize variables, input and output pin modes, and other libraries needed in the sketch. it is analogous to the function main

- loop(): after setup() function exits (ends), the loop() function is executed repeatedly in the main program. it controls the board until the board is powered off or is reset. it is analogous to the function while(1)

Power (usb / barrel jack)

Every arduino board needs a way to be connected to a power source. The arduino uno can be powered from a usb cable coming from your computer or a wall power supply (like this) that is terminated in a barrel jack. in the picture above the usb connection is labeled (1) and the barrel jack is labeled (2). the usb connection is also how you will load code onto your arduino board. more on how to program with arduino can be found in our installing and programming arduino tutorial. note: do not use a power supply greater than 20 volts as you will overpower (and thereby destroy) your arduino. the recommended voltage for most arduino models is between 6 and 12 volts.

Pins (5v, 3.3v, gnd, analog, digital, pwm, aref)

The pins on your arduino are the places where you connect wires to construct a circuit (probably in conjunction with a breadboard and some wire. they usually have black plastic headers that allow you to just plug a wire right into the board. the arduino has several different kinds of pins, each of which is labeled on the board and used for different functions.

- gnd (3): short for „ground". there are several gnd pins on the arduino, any of which can be used to ground your circuit.

- 5v (4) & 3.3v (5): as you might guess, the 5v pin supplies 5 volts of power, and the 3.3v pin supplies 3.3 volts of power. most of the simple components used with the arduino run happily off of 5 or 3.3 volts.

- analog (6): the area of pins under the „analog in" label (a0 through a5 on the uno) are analog in pins. these pins can read the signal from an analog sensor (like a temperature sensor) and convert it into a digital value that we can read.

- digital (7): across from the analog pins are the digital pins (0 through 13 on the uno). these pins can be used for both digital input (like telling if a button is pushed) and digital output (like powering an led).

• pwm (8): you may have noticed the tilde (~) next to some of the digital pins (3, 5, 6, 9, 10, and 11 on the uno). these pins act as normal digital pins, but can also be used for something called pulse-width modulation (pwm). we have a tutorial on pwm, but for now, think of these pins as being able to simulate analog output (like fading an led in and out). xxi

• aref (9): stands for analog reference. most of the time you can leave this pin alone. it is sometimes used to set an external reference voltage (between 0 and 5 volts) as the upper limit for the analog input pins.

Reset button

Just like the original Nintendo, the arduino has a reset button (10). pushing it will temporarily connect the reset pin to ground and restart any code that is loaded on the arduino. this can be very useful if your code doesn't repeat, but you want to test it multiple times. unlike the original Nintendo however, blowing on the arduino doesn't usually fix any problems.

Power led indicator

Just beneath and to the right of the word "uno" on your circuit board, there's a tiny led next to the word „on" (11). this led should light up whenever you plug your arduino into a power source. if this light doesn't turn on, there's a good chance something is wrong. time to re-check your circuit!

Tx rx leds

Tx is short for transmit, rx is short for receive. these markings appear quite a bit in electronics to indicate the pins responsible for serial communication. in our case, there are two places on the arduino uno where tx and rx appear -- once by digital pins 0 and 1, and a second time next to the tx and rx indicator leds (12). these leds will give us some nice visual indications whenever our arduino is receiving or transmitting data (like when we"re loading a new program onto the board).

Main Ice

The black thing with all the metal legs is an ice, or integrated circuit (13). think of it as the brains of our arduino. the main ice on the arduino is slightly different from board type to board type, but is usually from the at mega line of ic‟s from the at Mel company. this can be important, as you may need to know the ice type (along with your board type) before loading up a new program from the arduino software. this information can usually be found in writing on the top side of the ice. if you want to know more about the difference between various ic's, reading the datasheets is often a good idea. xxii

Voltage regulator

The voltage regulator (14) is not actually something you can (or should) interact with on the arduino. But it is potentially useful to know that it is there and what it‟s for. The voltage regulator does exactly what it says -- it controls the amount of voltage that is let into the arduino board. think of it as a kind of gatekeeper; it will turn away an extra voltage that might harm the circuit. of course, it has its limits, so don’t hook up your arduino to anything greater than 20 volts.

The arduino family

Arduino makes several different boards, each with different capabilities. In addition, part of being open source hardware means that others can modify and produce derivatives of arduino boards that provide even more form factors and functionality. if you’re not sure which one is right for your project, check this guide for some helpful hints. here are a few options that are well-suited to someone new to the world of arduino:

Ultrasonic sensor (hc-sr04)

- An ultrasonic sensor transmits pressure waves of sound energy at frequencies between 25 and 50 KHz, which is above the human audible range. Most of the ultrasonic sensors operate with pulse waveforms, which detect the presence of a vehicle and measure its speed.

PIN NUMBER	PIN NAME	DESCRIPTION
1	Vcc	The vcc pin powers the sensor, typically with +5v
2	Trigger	Trigger pin is an input pin. This pin has to be kept high for 10us to initialize measurement by sending us wave.
3	Echo	Echo pin is an output pin. This pin goes high for a period of time which will be equal to the time taken for the us wave to return back to the sensor.
4	Ground	This pin is connected to the ground of the system.

Hc-sr04 sensor features

- operating voltage: +5v
- theoretical measuring distance: 2cm to 450cm
- practical measuring distance: 2cm to 80cm
- accuracy: 3mm
- measuring angle covered: <15°
- operating current: <15ma
- operating frequency: 40hz you can buy hc-sr04 ultrasonic sensor from here. equivalent distance measuring sensors us transmitter receiver pair, ir sensor

module, if sensor pair, if analog distance sensor, hc-sr04 ultrasonic sensor – working.

Figure: As shown above the hc-sr04 ultrasonic (us) sensor is a 4 pin module, whose pin names are vcc, trigger, echo and ground respectively. This sensor is a very popular sensor used in many applications where measuring distance or sensing objects are required. the module has two eyes like projects in the front which forms the ultrasonic transmitter and receiver. the sensor works with the simple high school formula that

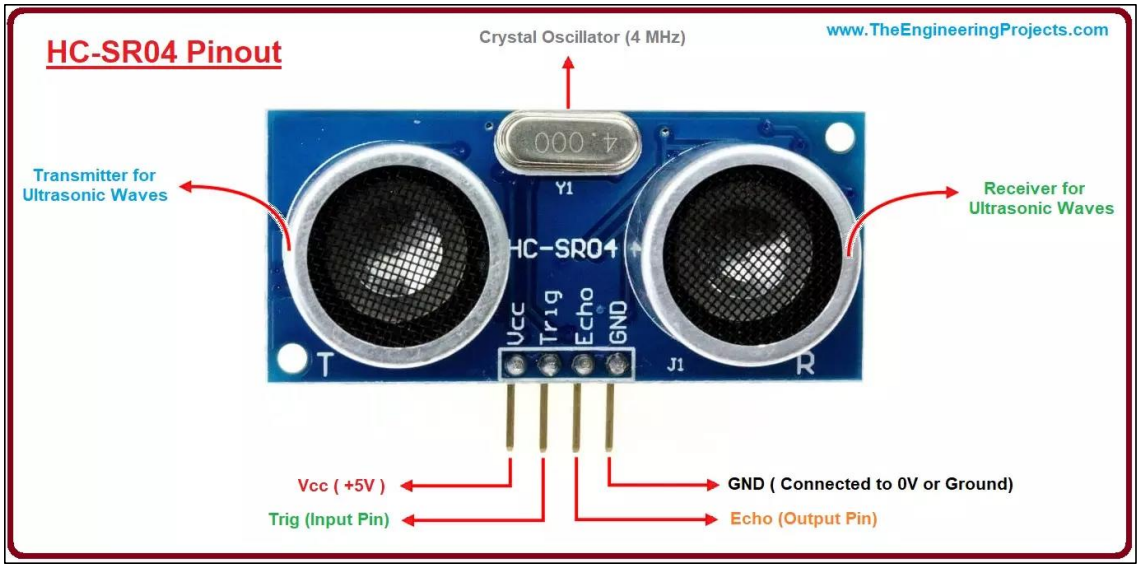


Figure no-2: Hc-sr04 ultrasonic sensor [2]

Distance = speed × time

The ultrasonic transmitter transmits an ultrasonic wave, this wave travels in air and when it gets objected by any material it gets reflected back toward the sensor this reflected wave is observed by the ultrasonic receiver module as shown in the picture below

Measurement Figure

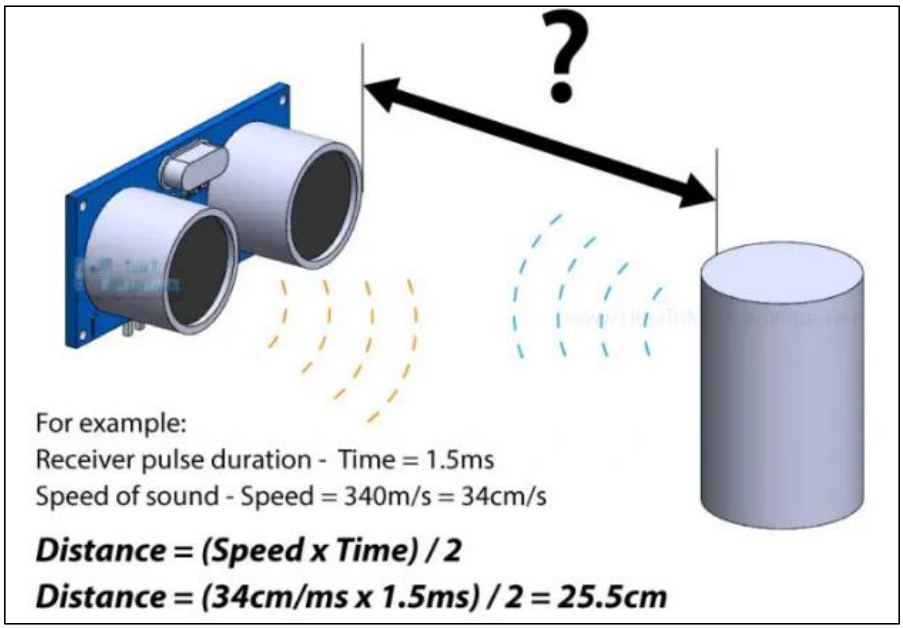


Figure no-2: Ultrasonic distance

Applications

- Used to avoid and detect obstacles with robots like biped robot, obstacle avoider robot, path finding robot etc.
- Used to measure the distance within a wide range of 2cm to 400cm
- Can be used to map the objects surrounding the sensor by rotating it
- Depth of certain places like wells, pits etc can be measured since the waves can penetrate through water

Board Prototype Circuit Bus Breadboard

Grid	: About 0.1inch (2.54mm)
Thickness	: About 1.5mm
Hole diameter	: Approx. 1.0mm
Dimension	: Approx. 4.8x13.4cm (W*L)
Material	: Bakelite

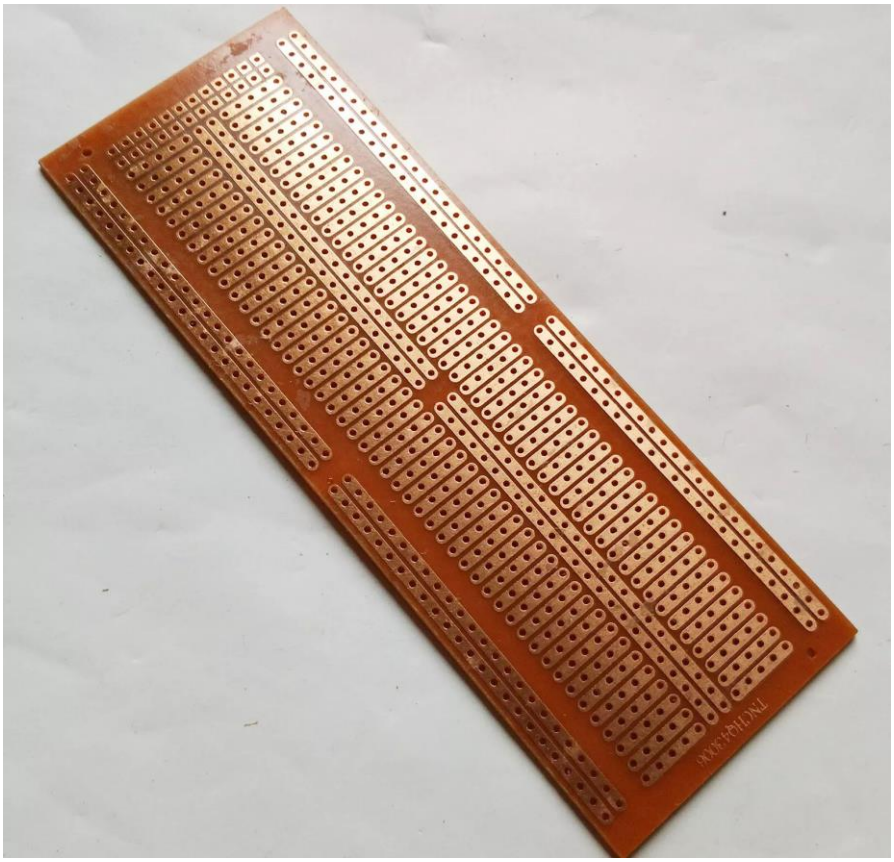


Figure no-4: Circuit bus [3]

Relay Module

Relay is one kind of electro-mechanical component that functions as a switch. The relay coil is energized by DC so that contact switches can be opened or closed. A single channel 5V relay module generally includes a coil, and two contacts like normally open (NO) and normally closed (NC). This article discusses an overview of the 5V relay module & its working but before going to discuss what is relay module is, first we have to know what is relay and its pin configuration.

The features of the 5V relay include the following.

- Normal Voltage is 5V DC
- Normal Current is 70mA
- AC load current Max is 10A at 250VAC or 125V AC
- DC load current Max is 10A at 30V DC or 28V DC
- It includes 5-pins & designed with plastic material
- Operating time is 10msec

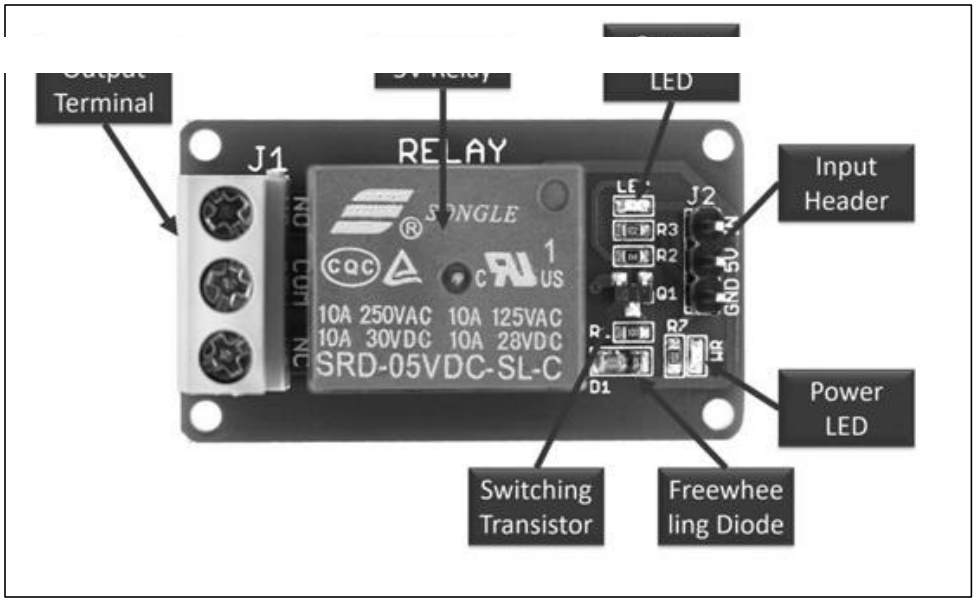


Figure no-5: Relay Module [4]

Relay

A 5V relay is coated with blue color plastic material. For both AC & DC loads, the utmost operating voltage & current are also displayed on the relay. This relay operates with 5V, so it is called a 5V relay.

Output Terminal

The output terminal of the relay module is located at the left-hand side, used to fix an AC/DC load & AC/DC i/p power source. Every o/p connector's terminal is connected through NC, COM pins & NO of the relay.

The relay module consists of screws that are used to connect wires & cables. The max current supported by this module is 10A & the max contact voltage is 250V AC & 30V DC. Thick main cables are mainly used whenever high voltage & current load is used.

Status Led

Status LED is connected by using a current limiting resistor that is located on the top right side of the relay module. So this LED illustrates the relay status by activating the relay & coil through a signal pin. The DC supplies throughout a relay coil.

Power Led

Power LED shows the condition of the power source that is connected through the single channel module. If we provide the above 5V source toward both the pins of the module like Vcc & GND, the LED will be damaged due to high voltage.

Input Connector

The input connector is located on the right side of the module. This connector is mainly used to supply a 5V power supply & input signal. In addition, it also supplies power supply toward the power LED, relay coil & status LED.

Water pump

DC 12V Solar Powered Water Pump Motor 700L/H Brushless Magnetic Submersible.

DC submersible pumps, on the other hand, are well-suited for remote locations, off-grid systems, and applications where a reliable DC power source is available, such as solar-powered irrigation systems or water supply in rural areas.

Feature

- Specifications: 10L/min 600L/H 160GPH, 16ft (5m), DC12V, 1.1A, 8W, come without plug (bare wire).
- Can used in computer cooling system, Garden fountain, Aquarium, car cooling, humidifier, air conditioner, other cooling and circulation systems.
- Brushless, permanent magnetic rotor, maintenance-free. Stator and circuit board sealed by epoxy resin.
- Super long working life (more than 40,000 hours).
- Small size, high efficiency, low consumption, low noise (less than 38db).

Specification

- Max flow rate: 10L/min 600L/H 160GPH
- Max lift height: 16ft (5m)
- Noise: <40dB (most 38dB)
- Power consumption: 8 W
- Rated voltage: 12V DC
- Rated current: 1.1A
- Power supply: Solar panel; DC electric source; battery
- Max working temp: 50 C/122F
- Fluids: Water, oil, gasoline, acid and alkali solution
- Waterproof level: IP68

Pump Figure

Shafi Water Pump Brand: Shafi Color: Black

Material: ABS

Solar Panel: 12V / 5W

Voltage: 12V DC

Maximum Rated Current: 1.2A

Power: 8W

Max Flow Rate: 10 L/Min

Max Water Head: 5M

Max. Water Height: 40-100cm

Max. Lift: 150cm / 59in

Waterproof Grade: IP65



Figure no-6: Water pump [5]

Transfers water from the reserve tank to the overhead tank when necessary

16x2 Lcd Display With Iic/I2c Interface – Blue

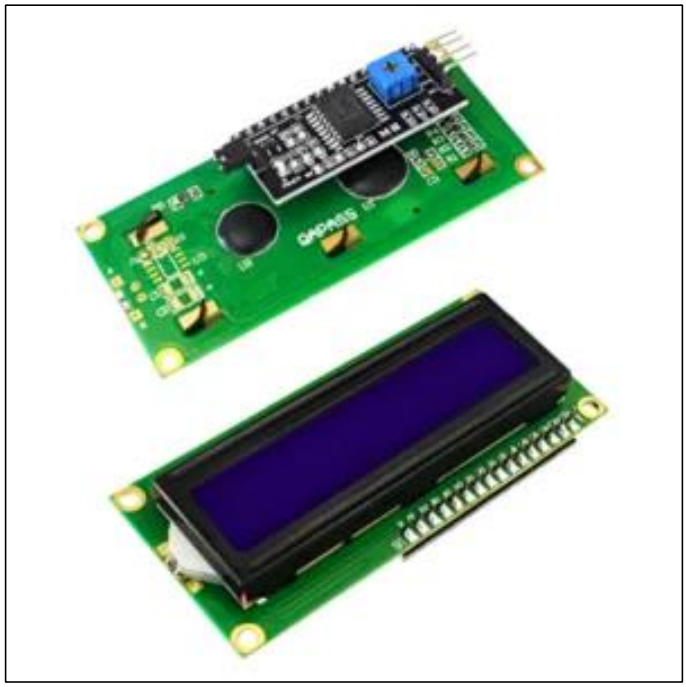


Figure no-7: LCD Display [6]

Whenever we work with embedded system we need a reliable output device with the help of which we get the required information, now this problem is solved with the introduction of 16.

character by 2 i.e. 16X2 LCD Display with IIC/I2C interface – Blue. 16×2 LCD is an alphanumeric display that can show up to 32 characters on a single screen. You can display more characters by scrolling the texts one by one. We already know that to connect LCD Display directly with the Arduino using 4bit and 8bit modes will utilize many numbers of GPIO Pins of our Arduino or other boards and we would have to end up with less number of pins for other sensors and actuators.

To overcome this problem we use LCD I2C backpack with our LCD. This I2C Backpack uses PCF8574 Remote 8 bit I/O Expander. It translates the data received from the I2C Bus into Parallel data that is needed for the LCD Display.

I2c – inter-integrated circuit

Inter-integrated Circuit (in short I2C) is a two-wire short distance communication protocol. You can use multiple slave devices in the same two wires with one or more master controllers. You may wonder how does the master identifies which slave does the data to be sent. In I2C the external devices have an I2C address for different external devices like LCD Backpack, OLED Display, etc. By using the address the data is sent to the specific device connected on the same I2C Bus.

The message is broken into two frames and sent serially via the I2C Bus. The first frame contains the address, once the address matches with any device on I2C bus, that device will send an acknowledge signal to the master. After receiving the acknowledgment from the slave the data bits are sent. By this method an I2C bus works.

16×2 lcd display

A 16×2 LCD means it can display 16 characters per line and there are 2 such lines. In this LCD each character is displayed in the 5×7 pixel matrix.

Lcd pin out

There are totally 16 pins in an LCD Display. You can use directly all the pins in 8-bit mode with Arduino or 12 pins using 4-bit mode. In this tutorial, we use the I2C module for LCD and multiplex it into just 4 pins. This pin details might not be useful while using I2C Method but this is the actual pin details of all the pins in LCD Display.

- Vcc – Power Supply (5v)
- Vdd/GND – Ground
- V0 – Brightness Control using Potentiometer
- RS – Register select. Specify what we are sending Command or Data. Sets to 0 for Command mode like set Cursor, LCD Clear, Turn OFF LCD. Set 1 for data mode like sending Data/Characters.

- R/W – Read/Write. Mostly we are writing catalog/characters to the registers.
- E – Enable writing to Registers.
- D0 to D7 – Data pins. Send 4bit/8bit data to display characters. The Arduino library provides 4bit and 8bit mode. The data will be in ASCII format.
- A/LED+ – Anode (Backlight LED)
- K/LED- – Cathode (Backlight LED)
- I2C Pinout:
- The LCD I2C Backpack only has 4 Pins. They are
- GND – Ground
- VCC – 5V Power Supply
- SDA – Data Line
- SCK – Clock Line

The module has a contrast adjustment pot on the underside of the display. This may require adjusting for the screen to display text correctly.

Features

- Arduino IIC/I2C interface was developed to reduce the IO port usage on Arduino board
- I2C adapter allows flexibility in connections
- I2C Reduces the overall wirings.
- 16 characters wide, 2 rows

- White text on the Blue background
- Single LED backlight included can be dimmed easily with a resistor or PWM.

Jumper wire

Jumper wires are electrical wires with connectors on each end that are used to connect components of a circuit without soldering. They are used in prototyping, testing, and circuit design.

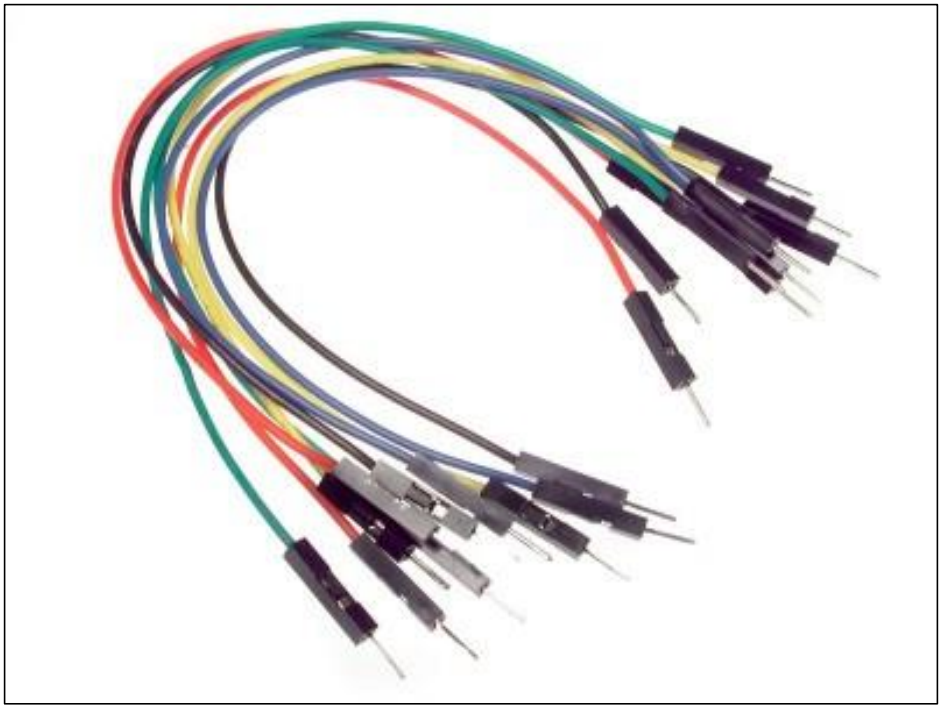


Figure no-8: Jumper Wire [7]

Types of Jumper Wires

Jumper wires typically come in three versions: male-to-male, male-to-female and female-to-female. The difference between each is in the end point of the wire. Male ends have a pin protruding and can plug into things, while female ends do not and are used to plug things into. Male-to-male jumper wires are the most common and what you likely will use most often. When connecting two ports on a breadboard, a male-to-male wire is what you'll need.

Features

- **Connectors:** Jumper wires can have male or female connectors.
- **Material:** Jumper wires can be made of copper or other materials.
- **Insulation:** Jumper wires can be insulated with PVC or other materials.
- **Color:** Jumper wires can be available in a variety of colors.
- **Length:** Jumper wires can be available in different lengths.

Uses

- Connecting components of a breadboard or other prototype circuit
- Connecting components of a printed circuit board
- Bypassing part of a circuit
- Establishing a connection between two points
- Determining if a part of a circuit is faulty or broken
- Short-circuiting and short-cutting (jumping) to an electric circuit
- Controlling electricity
- Stopping the operation of a circuit

Power supply

- I.T.E. Switching Power AC Adapter SHCY-SP1201000 12V, 1A White, 2 pin US Plug to power both the Arduino and the pump
- We supply that are used in a broad range of applications, such as computers, communications, consumer electronics, automotive electronics, cloud computing as well as healthcare.
- We are a world-leading provider of power supply. Based on the latest international standards, we developed and manufactured a power adapter, battery charger, and power supply.



Figure no-9: Adapter 12v [8]

2pin Switch

It is an SPST 2PIN Switch red mini Rocker Switch. Commonly it is used in a small home or office appliances, also AC and DC power ON-OFF switches, computers, and peripherals, medical instrumentation.

Switch Type: SPST
Contact resistance: <50mΩ
Insulation resistance: >100MΩ
Dielectric resistance: >600VAC / 1 min
Max Load 5A



Figure no-10: Push Bottom [9]

Led traffic light module 5v

Description

This is a mini-traffic light display module, high brightness, very suitable for the production of traffic light system model.

Small size.

Simple wiring.

Targeted.

Custom installation.

Specifications

Size: 56 x 21 x 11mm

Fixing hole: 3mm

Pitch: 15mm



Figure no-11: Traffic light Module [10]

Water Level Control Switch

A float switch is a type of level sensor, a device used to detect the level of liquid within a tank. It is a simple structure, easy to use level control device; moreover, it is neither a complicated circuit nor does it provide power. This switch has a mechanical switch than the average small sized switch.

As long as the material used is in the correct shape and nature of any liquid or pressure, you can control the temperature. Applications of this Switch include shipbuilding industry, generating equipment, petrochemical and food industry, water treatment equipment, dyeing, and finishing industry, and hydraulic machinery.



Figure no-12: Water level control switch [11]

Passive buzzer Module

A passive buzzer is a device that generates sound when an electrical signal is applied to it. It is called passive because it does not have an internal oscillator to generate sound on its own. Instead, it relies on an external signal from a microcontroller like Arduino to produce sound. The passive buzzer module is a small electronic component that contains a passive buzzer and some additional circuitry that makes it easier to use with Arduino.

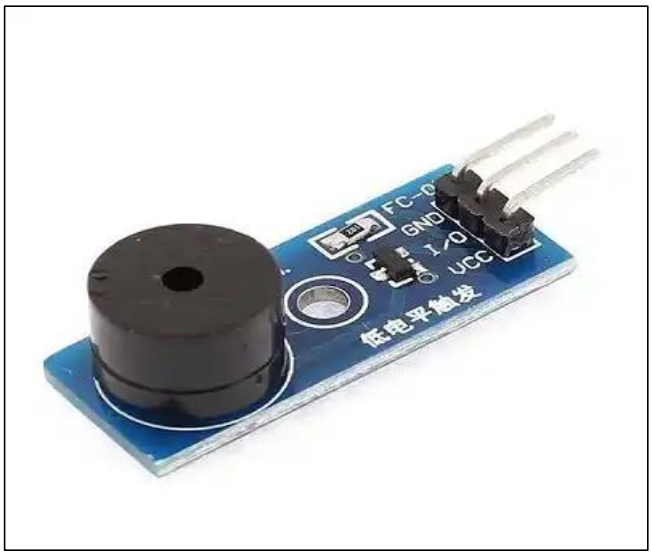


Figure no-14: Buzzer Module [13]

Hardware Components

- Arduino Uno R4 or R3 board * 1
- Passive Buzzer Module * 1
- Jumper Wires

Specifications of Active Buzzer Module

- It operates at a voltage range of 3.3V–5V DC.
- This buzzer module has a current rating of 30 mA at 5V DC.
- The resonance frequency of this buzzer is between 2500Hz ± 300Hz.
- It has a small size: 3.3 x 1.3 x 1.2 cm.

Chapter-3

Circuit Diagram

As shown in the water level controller circuit given below, Ultrasonic sensor module's "trigger" and "echo" pins are directly connected to pin 10 and 11 of arduino. A 16x2 LCD is connected with arduino in 4-bit mode. Control pin RS, RW and En are directly connected to arduino pin 7, GND and 6. And data pin D4-D7 is connected to 5, 4, 3 and 2 of arduino, and buzzer is connected at pin 12. 6 Volt relay is also connected at pin 8 of arduino through ULN2003 for turning on or turning off the water motor pump.

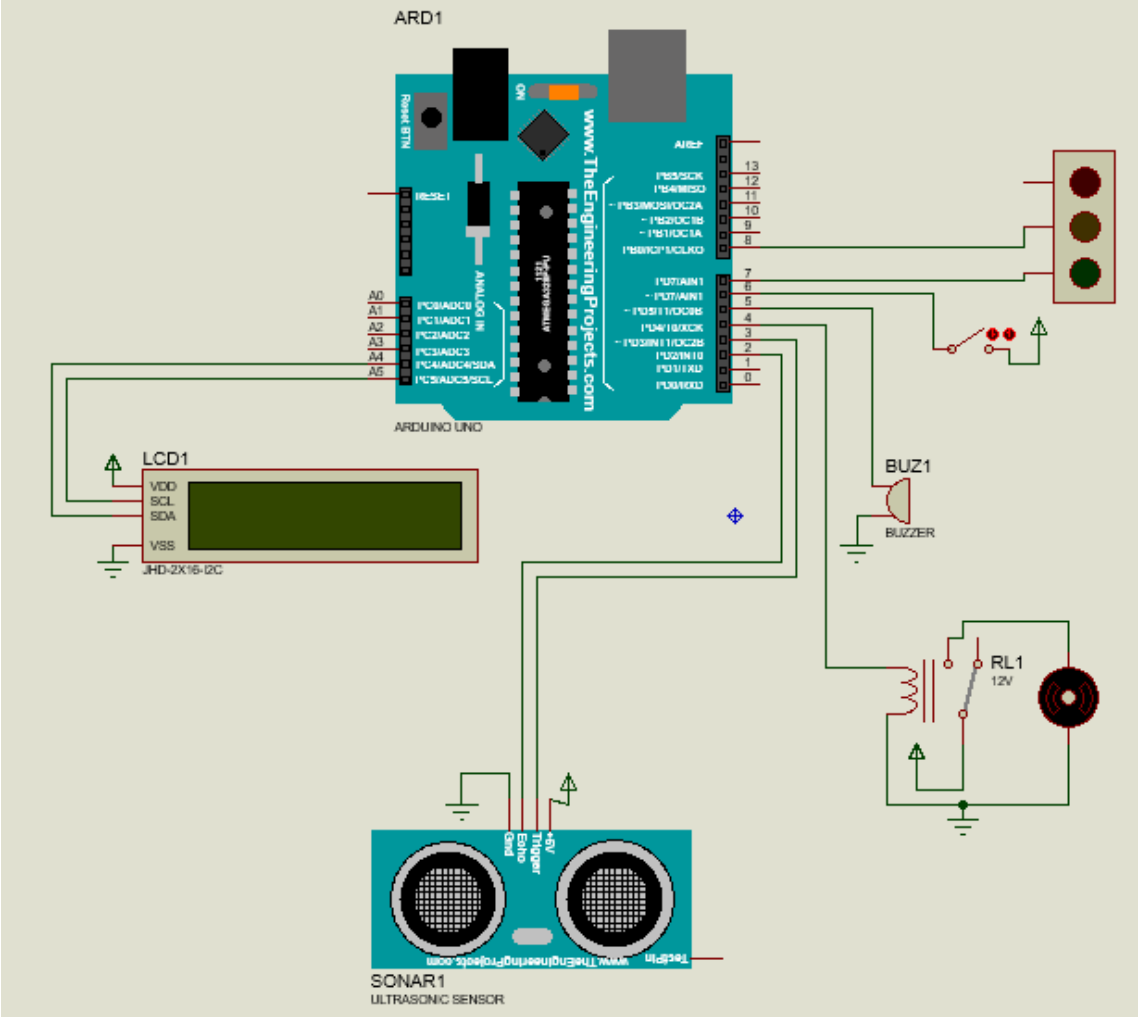


Figure no-15: Circuit Diagram [14]

In this circuit Ultrasonic sensor module is placed at the top of bucket (water tank) for demonstration. This sensor module will read the distance between sensor module and water surface, and it will show the distance on LCD screen with message “Water Space in Tank is:”. It means we are here showing empty place of distance or volume for water instead of water level. Because of this functionality we can use this system in any water tank. When empty water level reaches at distance about 30 cm then Arduino turns ON the water pump by driving relay. And now LCD will show “LOW Water Level” “Motor turned ON”, and Relay status LED will start glowing

Now if the empty space reaches at distance about 12 cm arduino turns OFF the relay and LCD will show “Tank is full” “Motor Turned OFF”. Buzzer also beep for some time and relay status LED will turned OFF.

A simple schematic diagram of the system can be shown here to illustrate the connections.

- **Ultrasonic Sensor (HC-SR04) Connections**
    - VCC: 5V
    - GND: Ground
    - Trig Pin: Connected to a digital pin on the Arduino (e.g., Pin 9)
    - Echo Pin: Connected to another digital pin on the Arduino (e.g., Pin 10)
- **Relay Module Connections**
    - VCC: 5V
    - GND: Ground
    - IN: Connected to a digital pin on the Arduino (e.g., Pin 7)
    - The relay contacts control the water pump circuit.
- **Water Pump**
    - The pump is powered through the relay and can be connected to a separate power source, depending on its voltage requirements.

Algorithm

1. Initialize the system

    - Set up the sensor pins.
    - Initialize the relay pin.
    - Optionally, initialize the LCD display.

2. Measure the distance

    - Trigger the ultrasonic sensor to send a pulse.
    - Measure the time for the pulse to return.
    - Calculate the distance using the speed of sound.

3. Calculate water level

    - Convert the measured distance to a percentage of the tank's height.

4. Control the pump

    - If the water level is below a predefined threshold, activate the relay to turn on the pump.
    - If the water level reaches the maximum level, deactivate the relay to stop the pump.

Chapter-4

Block Diagram

A block diagram for a water level indicator using an Arduino and ultrasonic sensor to control a pump between an overhead tank and a reserve tank would include: an ultrasonic sensor mounted on the overhead tank, sending ultrasonic pulses towards the water surface, with the reflected echo signal processed by the Arduino to calculate the water level; this data is then compared to pre-set thresholds to activate a relay which controls the pump, allowing water to transfer from the reserve tank to the overhead tank when the level drops below a set point; the system can optionally display the current water level on an LCD screen for monitoring. A Data Flow Diagram (DFD) is a graphical representation of the "flow" of data through an information system. DFD can also be used for the visualization of data processing. Automatic water level indicator and controller system uses two sensors at two levels of the tank, i.e. one at the higher level of the tank and the other at the lower level of the tank.

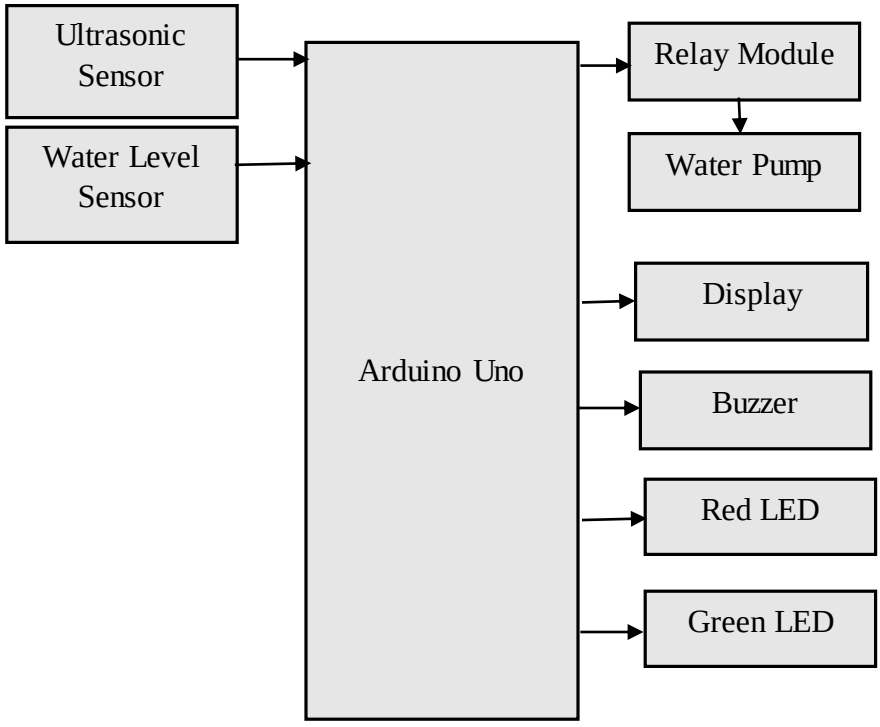


Figure no-16: Block Diagram [15]

System Design

The water level indicator consists of two primary parts:

Water Level Measurement

The ultrasonic sensor is mounted at the top of the overhead tank. It emits high-frequency sound waves, which bounce off the water surface.

The time taken for the sound waves to return is used to calculate the distance between the sensor and the water surface.

The distance is then converted into a water level percentage.

Pump Control

- The Arduino compares the measured distance with predefined threshold values to determine the water level.
- When the water level in the overhead tank drops below a certain threshold, the Arduino activates the relay, turning on the pump to transfer water from the reserve tank.
- Once the water level reaches a safe threshold, the Arduino deactivates the pump.

Working Principle

The working principle of the system is based on ultrasonic measurements. The HC-SR04 sensor continuously measures the distance between the water surface and the sensor. Using this information, the Arduino calculates the remaining water height. If the tank is low, a relay module turns on the pump to fill the overhead tank. The water level is continuously monitored, and the pump is turned off once the optimal level is reached.

Advantages

Automatic Operation: The system automatically turns the pump on when the overhead tank water level drops below a set threshold and off when it reaches a desired level, eliminating manual monitoring and preventing water shortages or overflows.

Water Conservation:

By precisely controlling the pump operation, the system minimizes water wastage and optimizes water usage.

Cost-Effective:

Compared to complex water level monitoring systems, an Arduino-based setup is relatively inexpensive to implement.

Real-Time Monitoring:

The system provides continuous feedback on the water level in the overhead tank, allowing for timely adjustments if necessary.

Easy To Install And Configure:

Arduino programming allows for straightforward setup and customization of water level thresholds.

Flexibility:

The system can be easily adapted to different tank sizes and water usage requirements.

Disadvantages:

Sensor Accuracy Limitations:

Ultrasonic sensors can be affected by factors like water quality, temperature, and tank geometry, potentially leading to inaccurate level readings.

Potential For Malfunction In Dirty Water:

If the water in the tank is heavily contaminated with debris, the ultrasonic sensor may not function properly.

Calibration Required:

To ensure accurate water level measurements, periodic calibration of the sensor is necessary.

Power Dependency:

The system requires a reliable power source to operate.

Limited Range:

Depending on the ultrasonic sensor used, the effective measurement range may not be suitable for very large tanks.

Potential For Electrical Issues:

Improper wiring or connection could lead to electrical problems within the

Benefits Of Water Level Indicators & Water Alarms

There are many benefits of water level indicators and water alarms including:

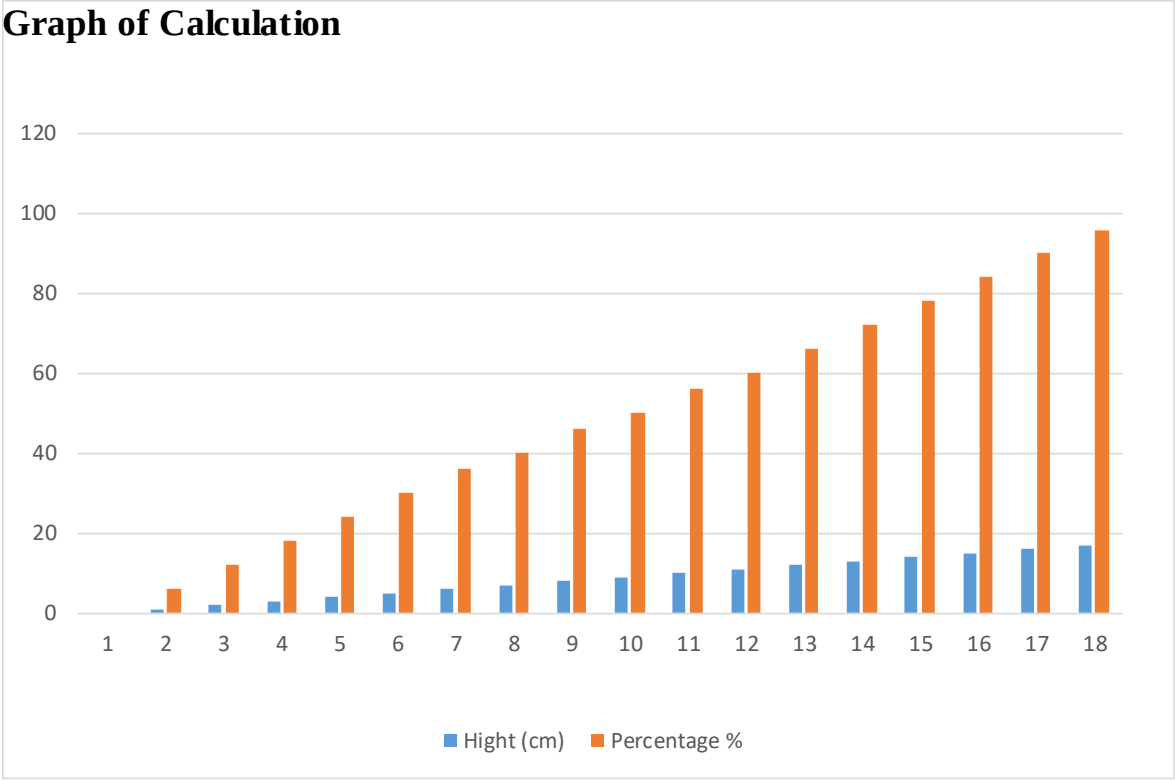
- Easy installation
- Minimal maintenance
- Sends an alert to let you know water is too high or too low
- Low & High alarms
- Compact design
- Automatically adjusts water levels
- Save money by using less electricity and water
- Can help avoid seepage of roofs and walls due to tanks overflowing
- Automatic operation saves you manual labor time
- Consumes a small amount little energy, perfect for on-going operations
- Indicates water levels in any type of storage tank or body of liquid
- A water alarm is loud so you can easily hear it.

Chapter-5

Calculation Table

Height Of Tank(cm)	Flow Time(second)	Percentage (%)
17	180	100
16	169.29	
15	158.71	
14	148.13	
13	137.55	
12	126.97	
11	116.39	
10	105.81	
9	95.23	50
8	84.65	
7	74.07	
6	63.49	
5	52.91	
4	42.33	
3	31.75	
2	21.17	
1	10.59	
0	00	0

Graph of Calculation



Result Analysis

Total Volume of Tank , $= \pi \times r^2 \times h$

$=3.1416*(12)^2*17$

$=7690.6\ cm^3$

$=76.906m^3$

It will take 3 Minutes to complete the water level in the tank of the overhead tank

1 minute =2.67 Ltr

2.67*3 minutes = 8.01 Ltr

8 Ltr of water can make 17 cm 100% flowers

$$\text{Average Of height} = \frac{1+2+3+4+5+6+7+8+9+10+11+12+13+14+15+16+17}{17}$$

= 9 cm of tank height 50%

Ultrasonic Sensor high accuracy is measured by (0% - 5.90%)

1 cm height water measured percentage, = 100/17 cm

= 5.90%

A 12V 5W DC Pump How much electricity use per minute?

Solution: Data,

V= 12V

P= 5W

t= 10.59s

We know that,

Law,

Power = Voltage * current =>I = P/V

=>I = 5/12

=> I ≈ 0.4167 A

Time Calculation 10.59s, = Total current * Time

= 0.4167 A * 10.59 Minutes

= 4.4129 A-s

Chapter-6

Conclusion

A Conclusion Automation of variety of things has been done in our developed environment. Where we can add multiplied to lesson inventions of what we invented and save out time. The water tank's water will be wasted because of its size and water will overflow if it fills. This leads more power intake as very high challenge at the maximum point of water inside of the tank we cannot be randomly guessed we can guess but it is not going to be right. So this results will help us to take more power as input as intake and that's the high challenge in the present. People additionally need to wait the time and make what they do their doing.. Thus we save both energy and power and time at the same time. From now on the computerized water stage indicator and controller the usage of Arduino and the whole equipment will be done without a doubt be beneficial on a large scale basics due to minimum level of necessary of the energy and additionally the system will be more simpler while making it simpler and to make it a bit more compatible for everyone to use it safely. B. Future Scope Here the automated water level indicator will be used at Arduino venture can be setup by us using PH sensors so that it will help us to alter the acidity in water which will be stored in the tank and the alkalinity of the water which will be stored inside the water tank.

Chapter-7

Program Code

```
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x3C, 16, 2);

#define trigPinA1          2
#define echoPinA1          3

#define relay              4
#define buzzer             5

#define water              6

#define g                  7
#define r                  8

void setup() {
  lcd.begin();
  lcd.backlight();
  Serial.begin(9600);
  pinMode(trigPinA1, OUTPUT);
  pinMode(echoPinA1, INPUT);

  pinMode(water, INPUT);

  pinMode(relay, OUTPUT);
```

```
  pinMode(buzzer, OUTPUT);
  pinMode(r, OUTPUT);
  pinMode(g, OUTPUT);

  digitalWrite(r, LOW);
  digitalWrite(g, LOW);
  digitalWrite(relay, HIGH);
  noTone(buzzer);
}
void loop() {
  int water_V = digitalRead(water);
  int dis = sennorA1();
  dis = map(dis, 5, 30, 0, 100);

  if (dis > 95) {
    digitalWrite(relay, HIGH);
    tone(buzzer, 800);
  }
  else {
    digitalWrite(relay, LOW);
    if (dis == 0) {
      tone(buzzer, 800);
    }
    else {
      noTone(buzzer);
    }
  }
  if (water_V == 1) {
```

```
    digitalWrite(r, LOW);
    digitalWrite(g, HIGH);
  }
  else {
    digitalWrite(r, HIGH);
    digitalWrite(g, LOW);
  }
  Serial.println(dis);

  lcd.setCursor(0, 0);
  lcd.print("Water Level :   ");
  lcd.setCursor(0, 1);
  lcd.print(dis);
  lcd.print("%   ");
}
long sennorA1() {
  digitalWrite(trigPinA1, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPinA1, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPinA1, LOW);
  long duration = pulseIn(echoPinA1,  HIGH);
  int distance = duration * 1.034 / 2;
  return distance;
}
```

Chapter-8

References

1. Arduino Uno R3 Board official website: https://docs.arduino.cc/tutorials/uno-rev3 /intro-to-board/
2. Hc-sr04 ultrasonic sensor documentation https://www.sciencedirect.com/topics/ Engineering/ultrasonic-sensor.
3. Bread Board (https://www.instructables.com/Using-Perfboard).
4. Relay Module Basics: [https://www.electronic tutorials] (https://www.electronic tutorials).
5. Water pump DC 12V Solar Powered Water Pump Motor 700L/H Brushless Magnetic Submersible.
6. 16x2 Lcd Display ith Iic/I2c Interface–Blue, https://store.roboticsbd.com/arduino-shield/1349-16x2-serial-lcd-module-display-for-arduino-assembled-robotics-bangladesh.html.
7. Jumper wire link, What Are Jumper Wires: Cooler, Types and Uses | Wiltronics
8. Adaptor Power Supply 12V Power Supply | Overview of 12V Power Supplies
9. Spst 2pin switch red mini push bottom SPST 2PIN Switch red mini Push Button-Calcutta Electronics.
10. Traffic Light 5V LED Module Top of Form 35.00 EGP, Traffic Light 5V LED Module | RAM Electronics website
11. Band, E. J., & Anyasi, F. I. (2014). Design of an automatic water level controller using mercury float switch. IOSR Journal of Electronics and Communication Engineering, 9(2), 16-21.
12. Getu, B. N., & Attia, H. A. (2016, December). Automatic water level sensor and controller system. In 2016 5th International Conference on Electronic Devices, Systems and Applications (ICEDSA) (pp. 1-4). IEEE.
13. Buzzer Module Interfacing with Arduino – Sound Code Examples, Buzzer Module Interfacing with Arduino Sound Code with Examples.

14. Zhang, G. (2013, May). Research on automatic water level detection technique and its application for open water-replenishing tank of central air conditioning. This report provides a comprehensive understanding of how the water level indicator works, and you can further elaborate on the technical details or add practical considerations based on your specific setup.

15. A block diagram for a water level indicator using an Arduino and ultrasonic sensor to control a pump between an overhead tank and a reserve tank would include: Our team member.

Future Recommendation

There are two sensors used in underground tank and the other in the overhead tank.

- If the underground tank is empty and full will give Signals
- And overhead 95% of the tanks can be 12 inch measurement

Chapter-9

Appendix

Real time are setup project all equipment.

