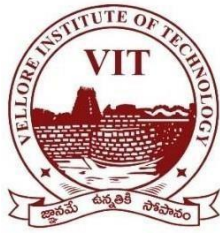


VIT-AP University, navolu, Amaravathi,
Andhra Pradesh-522 237, India



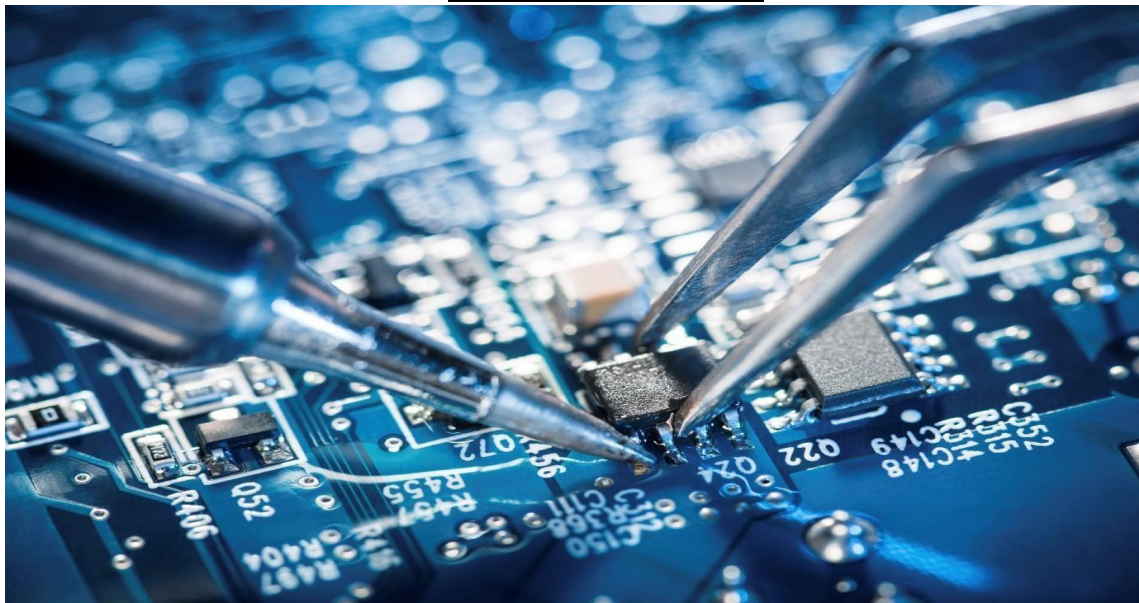
VIT-AP UNIVERSITY

AUTO HAND SANITIZING DEVICE

USING ARDUINO MODULE

Guided by Sir Mohammed iqbal

CLINICS REPORT



DONE BY

Naladala Navya- 21BCE7791

S.Hema sagar-21BCE7769

Nikhil Lekkala -21BCE7205

DMS Vikas-21BCE8639

K.Naveen Abhiram-21BCE7357

S.Moolya Sri-21BCE8087

ACKNOWLEDGMENT

The satisfaction that accompanies the successful completion of this project would be complete without the mention of the people who made it possible, without the constant guidance and encouragement would have made efforts go in vain. I consider myself privileged to express gratitude and respect toward all those who guided us through the completion of this project

We convey our special thanks to our project guide prof. Mohammed Iqbal of the Computer science and Engineering Department for providing encouragement, constant support, and guidance which was a great help to complete our project successfully

ABSTRACT PAGE(SUMMARY OF THE PROJECT)

Arduino is an open-source electronics creation platform, which is based on free, flexible, and easy-to-use hardware and software for creators and developers. This platform allows you to create different types of single-board microcomputers to which the community of creators can give different types of use

The Arduino IDE uses C++ but of course, the physical environment is limited so not all C/C++ features can be used the Arduino environment has helper functions to enable you to use the hardware easily and the Serial monitor for input/output.

TABLE OF CONTENT

1. INTRODUCTION

1.1 purpose of Arduino

1.2 System Overview

2. BACKGROUND

3. Problem Definition

4. Objectives

5. Methodology/Procedure

6. Results and Discussion

7. Conclusion and Future Scope

8. References

9. Codes in Appendix

LIST OF FIGURES

1. Common Components of Arduino Boards

2. Components

- I. Arduino module
- II. IR sensors
- III. Jumper wiring
- IV. Thermal sensor
- V. Depth sensor
- VI. Buzzer
- VII. LCD Display

3. Circuit Schematic for the module

4. experiment Layout

5. processing methodology

Introduction

An Arduino board contains a microcontroller that can be programmed to sense and control devices in the physical world. The microcontroller can interact with a large variety of components such as LEDs, motors, and displays. Because of its flexibility and sustainability, Arduino has become a popular prototyping development board that is widely used across the world.

Arduino has many uses from coding simple projects to complex systems and demonstrating real-life applications. The Arduino IDE is used to write programs and compile them to be loaded on the board. This supports a common programming language, C++.

1.1 Purpose of Arduino

Arduino comprises both a physical programmable circuit board (commonly known as a microcontroller) and programming software, or IDE (Integrated Development Environment) that can be run on a PC, and used to compose and transfer PC code to the circuit board. It can be done by using the Arduino programming language (based on Wiring), and the Arduino Software (IDE), based on Processing. Unlike other programmable circuit boards, the Arduino does not require different equipment (called a software engineer) to upload code to the circuit board, one can essentially utilize a USB link. Also, the Arduino IDE utilizes a rearranged rendition of C++, making it simpler to figure out how to program.

In a word, Arduino makes the functions of the microcontroller into a more accessible package. The Uno is one of the more prevalent boards in the Arduino family and an extraordinary option for the beginner

Problem Definition

As a part of the pandemic whole nation suffered a lot but during this pandemic social distancing and frequent hand, sanitization were the key steps to counter this pandemic. But manually doing this activity may be contagious during this pandemic so we need to do all these by machine technology

1. Arduino Board not Recognized

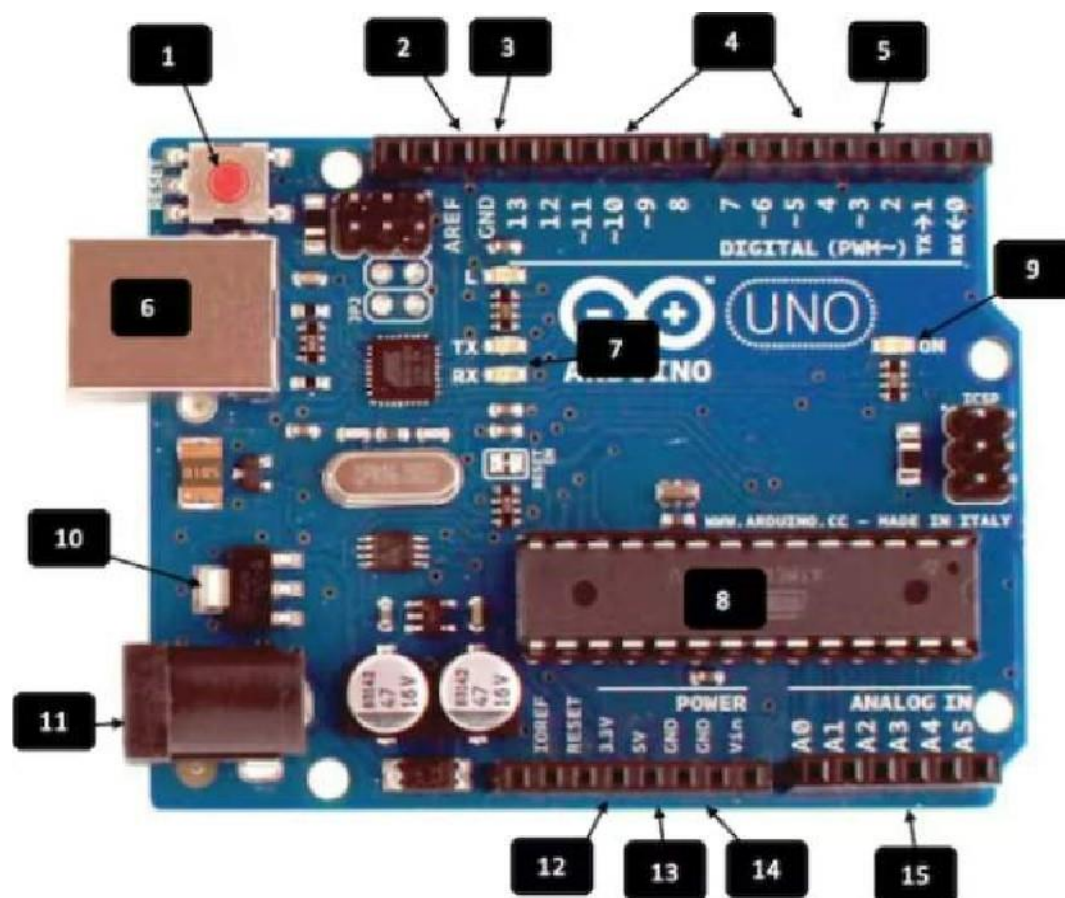
This refers to a situation where an Arduino board, connected to a computer is not recognized by the computer. When this happens, the board is usually not listed under the port lists of the Arduino IDE and is sometimes labeled USB 2.0 under the device manager.

2. The code doesn't start on Power Reset

This refers to cases where the Arduino does not run the sketch uploaded to it when powered On and in most cases, just reverts to the blink sketch associated with the bootloader.

3. Serial Port Already in Use

This is probably one of the easiest errors to resolve. It usually occurs when you try to upload code to an Arduino while the serial monitor is opened (this is no longer an issue if using recent IDE versions) or when you try to launch the serial monitor when the Arduino is talking to another software or device via the serial port. Essentially, it occurs when you try to use the Arduino serial port for two different things at the same time.



Objectives

1. Make the contactless temperature work using thermocouples, real-time digital simulators (RTDs), thermistors, and semiconductor integrated currents (ICs).
2. Connect jumper wires to each panel securely as they are delicate.
3. Checking power consumption, as high power consumption leads to damage of aurdino

4. Coad dumping to necessary part and make the other sensors work accordingly

Procedure

Step 1 — First you must have your Arduino board (you can choose your favorite board) and a USB cable. In case you use Arduino UNO, Arduino Duemilanove, Nano, Arduino Mega 2560, or Diecimila, you will need a standard USB cable (A plug to B plug), of body text.

Step 2 — Download Arduino IDE Software.

You can get different versions of Arduino IDE from the Download page on the Arduino Official website. You must select your software, which is compatible with your operating system (Windows, IOS, or Linux). After your file download is complete, unzip the file.

Step 3 — Power up your board.

The Arduino Uno, Mega, Duemilanove, and Arduino Nano automatically draw power from either, the USB connection to the computer or an external power supply. If you are using an Arduino Diecimila, you have to make sure that the board is configured to draw power from the USB connection, The power source is selected with a jumper, a small piece of plastic that fits onto two of the three pins between the USB and power jacks. Check that it is on the two pins closest to the USB port.

Step 4-Connect the Arduino board to your computer using the USB cable. The green power LED (labeled PWR) should glow.

Step 5 — Launch Arduino IDE.

After your Arduino IDE software is downloaded, you need to unzip the folder. Inside the folder, you can find the application icon with an infinity label (application.exe). Double-click the icon to start the IDE.

Step 6 — Open your first project.

Once the software starts, you have two options —

- Create a new project.

- Open an existing project example.

To create a new project, select File * New.

To open an existing project example, select File + Example + Basics Blink.

Here, we are selecting just one of the examples with the name Blink. It turns the LED on and off with some time delay. You can select any other example from the list.

Step 7 — Select your Arduino board.

To avoid any error while uploading your program to the board, you must select the correct Arduino board name, which matches the board connected to your computer

Step 8 — Select your serial port.

Select the serial device of the Arduino board. Go to the Tools Serial Port menu. This is likely to be COM3 or higher (COM1 and COM2 are usually reserved for hardware serial ports). To find out, you can disconnect your Arduino board and re-open the menu, the entry that disappears should be of the Arduino board. Reconnect the board and select that serial port

Step 9 — Upload the program to your board. Before explaining how we can upload our program to the board, we must demonstrate the function of each symbol appearing in the Arduino IDE toolbar.

Results and discussion

1. It comes with an open-supply hardware feature that permits users to develop their kit
2. The software of Arduino is well-suited to all kinds of in-operation systems like Linux, Windows, Macintosh, etc.

It also comes with an open supply software system feature that permits tough software system developers to use the Arduino code to merge with the prevailing programing language libraries and may be extended and changed

4. For beginners, it is very simple to use.

Conclusion and future scope

we use an Ultrasonic distance sensor, Servo motor, and Arduino board. here I am using Arduino Uno. You can also use any other microcontroller. When we place our hand in front of the distance sensor, it will help the Arduino to measure the distance from the sensor to the object (here the hand). if the object is in the desired range, Arduino will write the servo to 180. The Servo motor is mounded on the hand sanitizer bottle. And the trigger of the bottle is connected to the servo by a thread. When the servo motor rotates, the trigger will press.

Add a The temperature sensor in Arduino converts the surrounding temperature to voltage. It further converts the voltage to Celcius, Celcius to Fahrenheit, and prints the Fahrenheit temperature on the LCL screen.

We will use a temperature sensor (TMP 36) of low voltage.

References

<https://create.arduino.cc/projecthub/akshayjoseph666/covid-19-automatic-hand-sanitizer-78cf6b>

https://www.tutorialspoint.com/arduino/arduino_temperature_sensor.htm

<https://circuitdigest.com/microcontroller-projects/interfacing-ir-sensor-module-with-arduino#>

<https://robocraze.com/products/leonardo-r3-board-compatible-with-arduino>

Codes in appendix

```
#include <Wire.h>
#include <Adafruit_MLX90614.h>
Adafruit_MLX90614 mlx = Adafruit_MLX90614();
#include <LiquidCrystal.h>
const int rs = 2, en = 3, d4 = 4, d5 = 5, d6 = 6, d7 = 7;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
const int pingPin = A1; // Trigger Pin of Ultrasonic Sensor
const int echoPin = A0; // Echo Pin of Ultrasonic Sensor
long duration;
int distance;
int ir=8;
int ir1=11;
int cf=12;
int cf1=13;
int mot=9;
int mot1=10;
int buz=A3;
void setup() {
  pinMode(cf,OUTPUT);
  pinMode(cf1,OUTPUT);
  pinMode(ir1,INPUT);
```



```
Serial.begin(9600);
Serial.println("Adafruit MLX90614 test");
mlx.begin();
lcd.begin(16,2);
pinMode(pingPin, OUTPUT);
pinMode(echoPin, INPUT);
pinMode(buz,OUTPUT);
lcd.setCursor(4,0);
lcd.print("WELCOME");
lcd.setCursor(0,1);
lcd.print("Sanitizer disp sys");
delay(2000);
lcd.clear();
pinMode(mot,OUTPUT);
pinMode(mot1,OUTPUT);
pinMode(ir,INPUT);
}
void loop() {
    delay(200);
    digitalWrite(pingPin, LOW);
    delayMicroseconds(2);
    digitalWrite(pingPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(pingPin, LOW);
    duration = pulseIn(echoPin, HIGH);
    distance=duration*0.034/2;
    int t=(mlx.readObjectTempF());
    int ob=digitalRead(ir);
    int ob1=digitalRead(ir1);
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("T:");
    lcd.setCursor(2,0);
    lcd.print(t);
    lcd.setCursor(8,0);
    lcd.print("O:");
    lcd.setCursor(10,0);
    lcd.print(ob);
    lcd.setCursor(0,1);
    lcd.print("L:");
    lcd.setCursor(2,1);
    lcd.print(distance);
    lcd.setCursor(8,1);
    lcd.print("D:");
```

```
lcd.setCursor(10,1);
lcd.print(ob1);
if(ob==0)
{
    digitalWrite(mot,1);
}
else
{
    digitalWrite(mot,0);
}
if((distance>15) || (t>100))
{
    digitalWrite(buz,1);
}
else
{
    digitalWrite(buz,0);
}
if(ob1==0)
{
    digitalWrite(cf,1);
    digitalWrite(cf1,0);
}
else
{
    digitalWrite(cf,0);
    digitalWrite(cf1,0);
}
}
```

Final Demo

