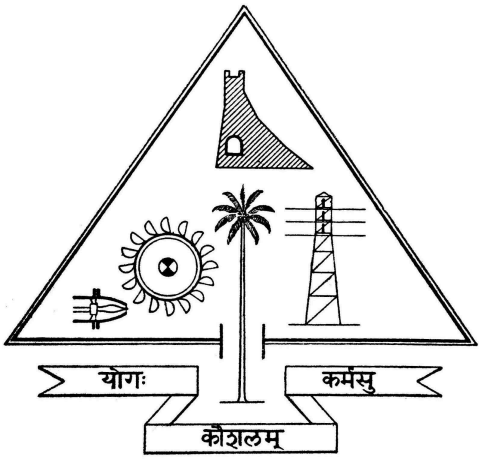
DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

GOVERNMENT ENGINEERING COLLEGE THRISSUR



PROJECT REPORT

on

SMART MEDICINE BOX

Submitted by

MALAVIKA K AJITH (TCR19EC037)

SANGEETHA KRISHNAKUMAR (TCR19EC048)

SIVAPRASAD A S (TCR19EC051)

VAISHNAV K S (TCR19EC056)

Guided by

Prof.Riyas K K

Professor

DEPT.OF ECE

GEC THRISSUR

SMART MEDICINE BOX

MINI PROJECT REPORT

Submitted by

MALAVIKA K AJITH (TCR19EC037)

SANGEETHA KRISHNAKUMAR (TCR19EC048)

SIVAPRASAD A S (TCR19EC051**)**

VAISHNAV K S (TCR19EC056**)**

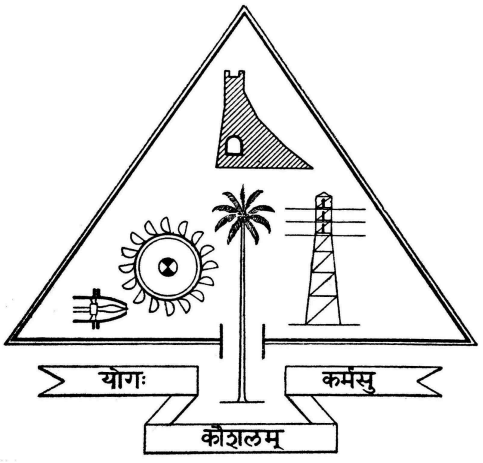
to
The APJ Abdul Kalam Technological University
in partial fulfillment of the requirements for the award of the Degree

of

Bachelor of Technology

in

Electronics and Communication Engineering



Department of Electronics and Communication Engineering
Government Engineering College Thrissur
JULY 2022

DECLARATION

We undersigned hereby declare that the project report (“SMART MEDICINE”), submitted for partial fulfillment of the requirements for the award of the degree of Bachelor of Technology of the APJ Abdul Kalam Technological University, Kerala is a bonafide work done by us under supervision of Prof. Riyas K K. This submission represents our ideas in our own words and where ideas or words of others have been included, We have adequately and accurately cited and referenced the original sources. We also declare that we have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in our submission. We understand that any violation of the above will cause disciplinary action by the institute and/or the University and can also evoke penal action from the sources that have thus not been properly cited or from whom proper permission has not been obtained. This report has not previously formed the basis for the award of any degree, diploma, or similar title of any other University.

Place: Thrissur

Date:

Signature:

Malavika K Ajith

Signature:

Sangeetha Krishnakumar

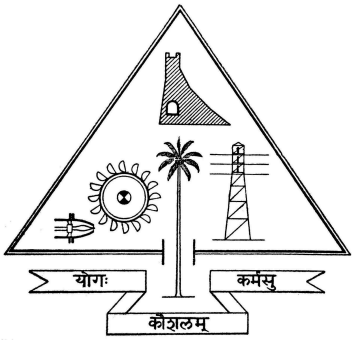
Signature:

Sivaprasad A S

Signature:

Vaishnav K S

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
GOVERNMENT ENGINEERING COLLEGE, THRISSUR



CERTIFICATE

This Is To Certify That The Report Entitled, **SMART MEDICINE BOX** submitted by **MALAVIKA K AJITH, SANGEETHA KRISHNAKUMAR, SIVAPRASAD A S, VAISHNAV K S ,** to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the Degree of Bachelor of Technology in Electronics and Communication Engineering is a bonafide record of the project work carried out by them under our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

Internal Supervisor

RIYAS K K
Professor
Govt.Engineering College
Thrissur

Head of The department

Dr. A R JAYAN
HOD & Professor
Govt.Engineering College
Thrissur

DEPARTMENT VISION

To become a nationally acclaimed Department of higher learning and research that will serve as a source of knowledge and expertise in Electronics & Communication Engineering.

DEPARTMENT MISSION

1. To provide quality education in the area of Electronics and Communication Engineering, to produce innovative and ethically driven professionals adept at dealing with a globally competitive environment, for the welfare of the nation.

2. To inculcate inquisitiveness in young graduates thereby persuading them to undertake research in emerging areas of Electronics and Communication Engineering.

Acknowledgement

We wish to record our indebtedness and thankfulness to all those who helped us prepare this project titled Smart Wearable for Patients and present it in a satisfactory way. First and foremost we thank Dr. A R Jayan, Head Of the Department, Dept of Electronics and Communication Engineering for giving us the opportunity to present this project.We extend our gratitude to Prof. Riyas K K,Professor, Dept of Electronics and Communication Engineering for his valuable guidance, encouragement and co-operation during the course of this project and its presentation.It was his support which resulted in successful presentation of this project. Finally, we would like to extend our sincere gratitude towards teachers, friends of Dept of Electronics and Communication Engineering and our family members who have always been helpful.

GEC Thrissur

Date :

Malavika K Ajith

Sangeetha Krishnakumar

Sivaprasad A S

Vaishnav K S

ABSTRACT

It is a purely microcontroller-based device meant for old-aged people who are living in their homes alone. As per the reports, In the modern world, the number of senior citizens living alone is increasing at a faster rate. So due to old age difficulties, some may forget to take medicine on time and it causes severe health problems. Existing internet-based reminder systems can be manipulated easily and it may cause irregularities in the intaking of medicines and thereby risks the lives of our senior citizens. So we designed a circuit that alerts the person at the time when he/she has to take the medicine and the robotic arm dispenses the medicine from the medicine box to the compartment from where the person can take the medicine. Our circuit mainly consists of Arduino mega 2560 microcontroller, RTC module, LCD screen, push button, buzzer, and robotic arm.

CONTENTS

LIST OF FIGURES

ABBREVIATIONS

RTC - Real Time Clock

NEMA -The National Electrical Manufacturers Association

LCD - Liquid Crystal Display

SDA - Serial DATA

SCL - Serial Clock

EEPROM-Electrically Erasable Programmable Read-Only Memory

SCARA- Selective Compliance Assembly Robot Arm

PWM-Pulse Width Modulation

AC-Alternate Current

DC-Direct Current

IoT-Internet of Things

SRAM-Static Random Access Memory

CHAPTER 1

INTRODUCTION

1.1 GENERAL BACKGROUND

As young minds are migrating to different places in order to pursue their career, most parents are becoming alone in their homes. The number of senior citizens living alone is increasing at a faster rate. So due to old age difficulties, some may forget to take medicine on time and which causes irregularities in medicine intake and thereby causes severe health problems. Our motive to work on this project is the importance of maintaining the health of old-aged people living in their homes alone.

1.2 OBJECTIVES

- ☐ Assist old aged people
- ☐ Reduce the burden on caretaker
- ☐ Provide medicine on time
- ☐ Avoid irregularity in taking medicine
- ☐ Keeping people healthy

1.3 SCOPE

The caretakers of old aged people may not be able to assist them all time. So our circuit system helps the old aged people to take medicine on time and stay healthy.

CHAPTER 2
LITERATURE SURVEY

2.1 EXISTING TECHNOLOGIES

2.1.1 IOT-BASED SMART MEDICINE DISPENSER

It is an IOT-based device to control and supervise the medication intake of elderly people

Device FEATURES:

- Smart medicine dispenser for solid medications
- Mobile application for its configuration and management

2.1.2 SMART PILL DISPENSER

This device helps patients and caretakers track medication compliance, reducing guesswork and stress while improving new levels of control and improved patient-doctor communication

Device FEATURES

- Tracks and reports medication intake
- Send alerts
- Order refills
- Improves adherence to strict medication timing

CHAPTER 3

ALARM SYSTEM

Here we made a simple Medicine Reminder using Arduino which reminds us to take medicines 1 or 2 or 3 times a day. The time slot can be selected using push buttons. Also, it shows the current Date and Time. We will further extend it to an IoT project incoming articles where an email or SMS notification will be sent to the user. This medication reminder can also be integrated with the Patient Monitoring System.

Components Required for Automatic Medicine reminder using Arduino are Arduino Mega, RTC DS3231 module,16x2 LCD Display, Buzzer, breadboard, Push Buttons, Jumper Wires

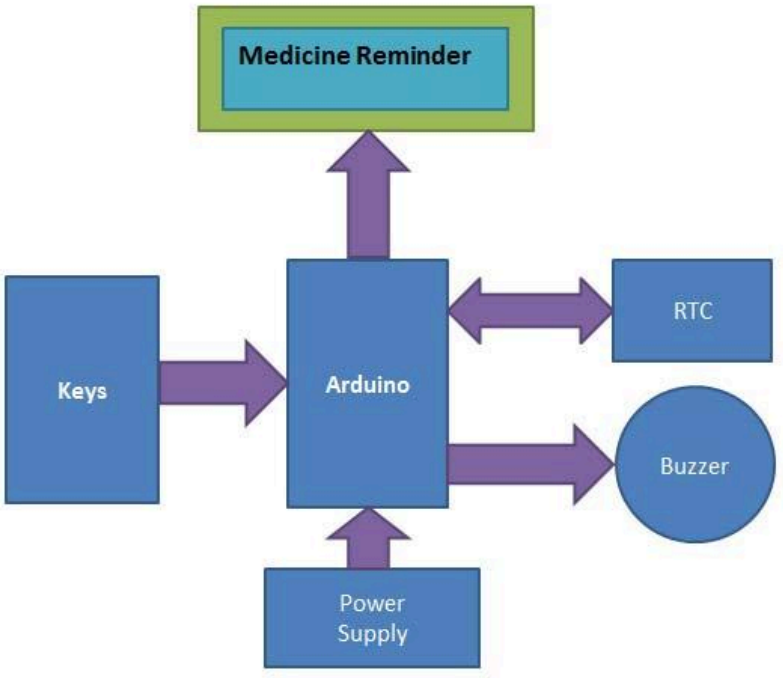


Fig. 3.1: Block Diagram of Arduino-based Medicine Reminder System

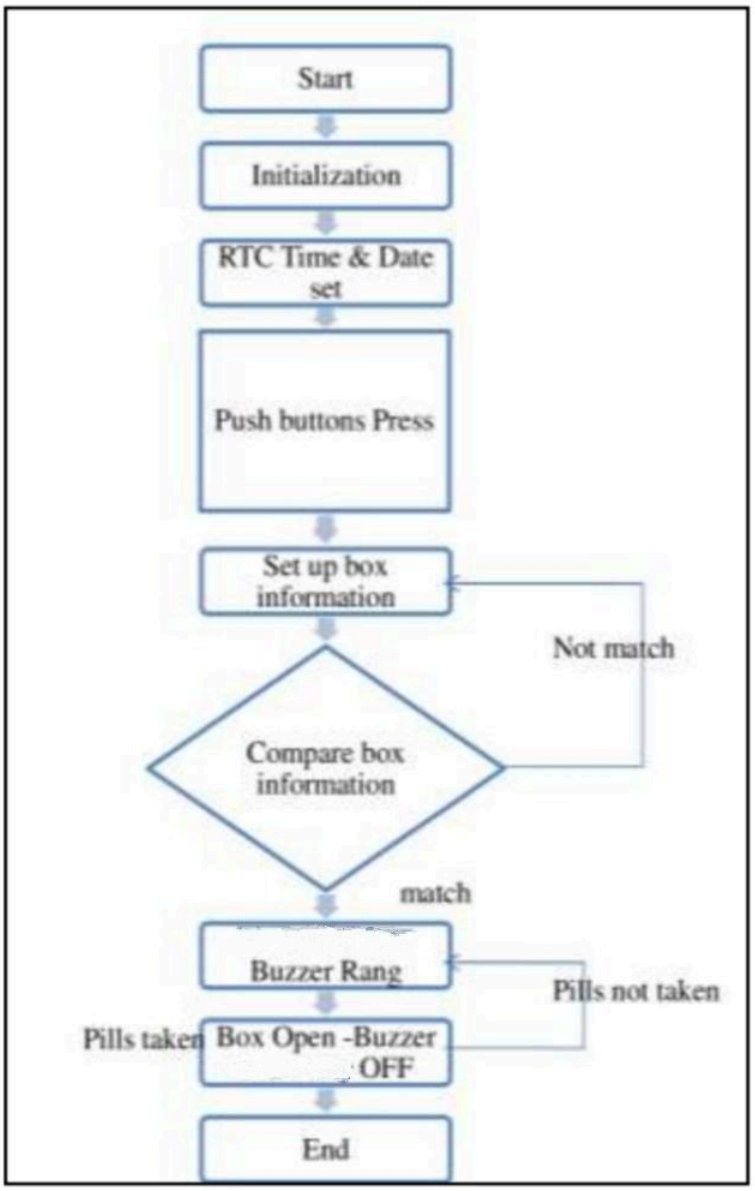


Fig 3.2 Flowchart of medicine reminder system

In this system, we have used Arduino for controlling the whole system. Working on this project is very simple. In this system ds1307 real-time clock, the chip is used for running the time accurately and to prevent the time after light failure by using a 3-volt li-on battery connected with this real-time clock chip at pin number 3. SDA and SCK pin of real-time clock chip is directly connected with SDA and SCK pin of Arduino (A5 and A4) respectively.

When we start this system, the real-time clock runs the time on a 16×2 LCD. And if we want to set the alarm time for medication we have to press the set_mad buttons which are connected with pin number 8 of the Arduino. After pressing this button the LCD shows Set Time 1. And then we can select the time we want to set for medication by using the INC and Next button which is connected to pins 9 and 10 respectively of the Arduino. After set time 1, LCD shows set Time 2. Now using the previous process, I set the time again. And after the second time set, LCD shows again set time 3. And set this time like previous. In this system "Group medicine" indication (take group 1 medicine, take group 2 medicine and take group 3 medicine) is used instead of the medicine name. When any alarm occurs LCD indicates Group medicine 1, Group medicine 2, Group medicine 3.

The medication alarm time is also fed into Arduino's internal EEPROM to save from losing data after a light failure. And real-time is continuously checked with saved Arduino's internal EEPROM time. If any match occurs. LCD shows the medication group name and the buzzer starts beeping continuously. The buzzer is directly connected with pin number 34 of Arduino for medication time indication.

16×2 LCD's data pins D4, D3, D2, and D2 are connected with pins 26, 28, 30, and 32 of Arduino. And command pin RS and EN is directly connected with pin 22, and 24 of Arduino. RWpin of LCD is directly connected with the ground.

Programming

The programming part of this system is much simple. In this program some libraries are used given below:

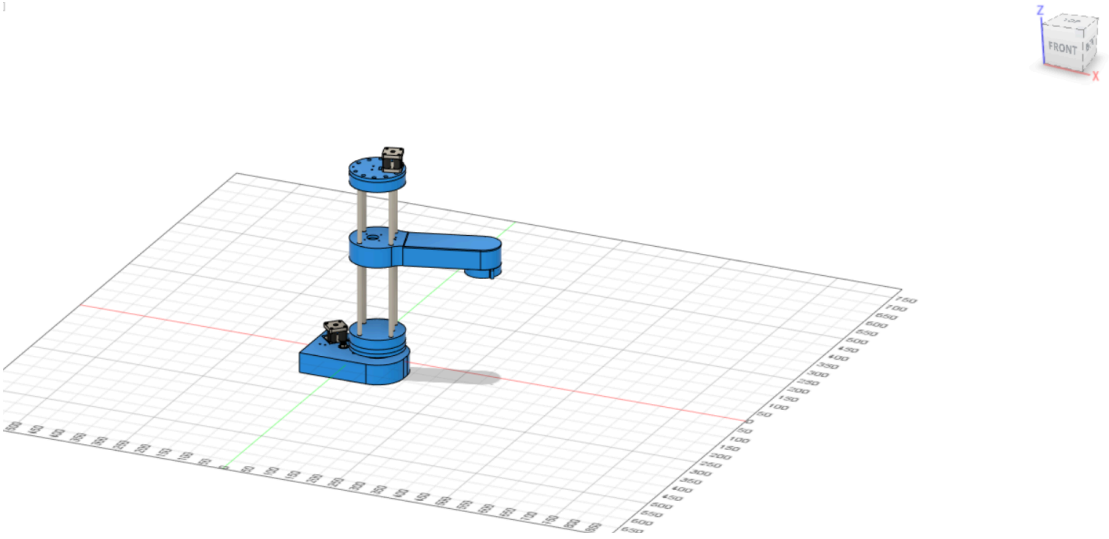
1. Wire. h: for I2C interfacing
2. RTClib.h: for RTC interfacing
3. LiquidCrystal: for 16X2 LCD interfacing
4. EEPROM.h: to access the internal Arduino's EEPROM for saving the alarm time.

CHAPTER 4

SCARA ROBOT

Here we show how to build an Arduino-based SCARA Robot. The robot has 3 degrees of freedom and it's driven by 3 NEMA 17 stepper motors. The brain of this SCARA robot is an Arduino UNO board which is paired with a CNC shield and four A4988 stepper drivers for controlling the stepper motors.

So let's get back to the model and explain how we came up with this design. Our goal for the robot was for most of the parts to be 3D printed. So, everything seen here can be 3D printed even on a 3D printer with a smaller printing bed. The GT2 pulleys are also 3D printable. We used parametric design to make them, so if needed we can easily change their sizes. We just have to change the number of teeth, and all dimensions will automatically update to make the pulley the proper size. For the first joint, we have a 20:1 reduction ratio, achieved in two stages with these custom-designed pulleys. The two GT2 belts I use here are closed loops of 200mm and 300mm length. The robot joints are composed of two thrust bearings and one radial bearing



Assembling the robot

Here's a list of components needed for assembling this Arduino-based SCARA robot.

- 4x Smooth rod shaft – 10mm 400mm, 1x Lead screw – 8mm 400mm, 1x Thrust ball bearing 40x60x13mm
- Stepper Motor – NEMA 17, A4988 Stepper Driver, Arduino CNC Shield, Arduino Mega 2560, DC Power Supply

SCARA Robot working

There are two methods for controlling robots in terms of positioning and orientation, and that's using forward or inverse kinematics. Forward kinematics is used when we need to find the position and orientation of the end-effector from the given joint angles.On the other hand, inverse kinematics is used when we need to find the joint angles for a given position of the end-effector. This method makes more sense in robotics as most of the time we want the robot to position its tool to a particular location or particular X, Y, and Z coordinates

So, we will use an Arduino MEGA board in combination with a CNC shield and three A4988 stepper drives. So finally, once we upload the code to the Arduino, we can run the processing program, connect the power and the scary robot will start moving to its home position.

CHAPTER 5

VACUUM PUMP

We have proposed a low-cost, simple homemade vacuum pump. In this project, we have made an attempt to build a simple vacuum cleaner by organizing the simple materials that are available in our surrounding environment. The simple materials we used here are a 3D printed vacuum bottle, PC cooling fan, and PVC pipes.

Vacuum cleaner technology is simple: create a low-pressure system that forces (not sucks, because sucking does not exist- but that is for another time) air into a tube where dust and small debris can be picked up and stored. In essence, a fan is connected to a tube.

Steps in making a vacuum machine are:

STEP-1 DESIGN OF VACUUM BOTTLE USING 3D PRINTING

A plastic bottle is a container for sucking the medicines from the medicine box and holding it till it is deposited in the tray. All the parts like a motor and filter are fitted inside the bottle.

STEP-2 MAKING OF VACUUM CLEANER FAN

Here we use 2 wire pc cooling fan. Only two wires come out of the fan controller, the positive and the negative. Giving power to the fan, it will rotate at full speed. Only two wires come out of the fan controller, the positive and the negative. Giving power to the fan, it will rotate at full speed, which helps in sucking the medicine and holds the medicine till the fan is OFF

STEP: 5 ATTACH FAN ASSEMBLY AND PLACE THE SUCTION PIPE

Insert the glue behind the plastic bottle to attach the fan. Also, insert the glue to attach the suction pipe to the cap of the bottle. We need a filter medium to collect the dust in the bottle.

STEP: 6 MAKING SUCTION PIPE

A suction pipe is the main design of a vacuum cleaner. If the Suction pipe we made is very long then it can not suck the dust inside it. So the pipe must be of average size.

STEP: 7 POWER SOURCE

Finally, wire up the motor and attach it to a plug assembly to connect it to the mains. The last thing is to connect the wires that are at the bottom of the battery for the electric supply. So the current starts to flow and the vacuum pump starts to work. Here we use a relay to control the ON and OFF vacuum pump.

CHAPTER 6
COMPONENTS

6.1 ARDUINO MEGA 2560

The Arduino Mega 2560 is a microcontroller board based on the ATmega2560. It has 54 digital input/output pins (of which 14 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with an AC to-DC adapter or battery to get started. The Mega is compatible with most shields designed for the Arduino Duemilanove or Diecimila.

Fig 6.1: Arduino Mega 2560

6.2 NEMA 17 STEPPER MOTOR

Fig 6.2: NEMA 17 STEPPER MOTOR

NEMA 17 stepper motor is a brushless DC electric motor that divides a full rotation down into equal steps. It is suitable for applications where precise movement is required.NEMA 17 is a hybrid stepper motor that rotates with a step angle of 1.8 degrees. Thus to complete one revolution, the shaft takes 200 steps. The term hybrid signifies that it is a combination of a permanent magnet and variable reluctance stepper motor. With dimensions, of 1.7 x 1.7 inches (42 x 42 mm), NEMA 17 steppers are engineered to provide more torque than smaller variants like NEMA 14 stepper motor. The NEMA 17 stepper motor consists of a stator and rotor. The rotor of the NEMA 17 motor is a permanent magnet with 50 teeth on its circumference. The stator is simply an electromagnet with 48 teeth.

6.3 CNC SHIELD

The CNC Shield can be used for the precision control of stepper motors. It uses open source firmware on Arduino to control 4 stepper motors using 4 A4988 Stepper drivers. It is designed as a shield and can plug on top of an Arduino requiring no external connections and wiring. There are 4 slots on the board for plugging in the stepper motor drive module which can drive 1 stepper motor each. Controlling each step stepper motor requires only two IO pins on the Arduino.

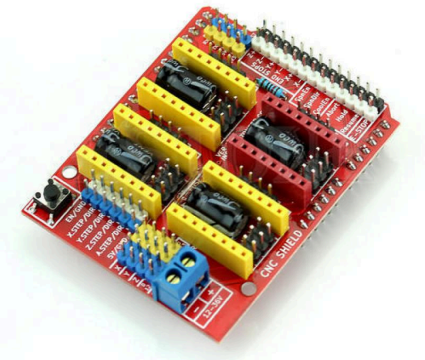


Fig 6.3: CNC SHIELD

6.4 A4988

The A4988 is a complete micro-stepping motor driver with a built-in translator for easy operation. It is designed to operate bipolar stepper motors in full-, half-, quarter-, eighth-, and sixteenth-step modes, with an output drive capacity of up to 35 V and ±2 A. The A4988 includes a fixed off-time current regulator which has the ability to operate in Slow or Mixed decay modes. It helps in easy control of the stepper motor.

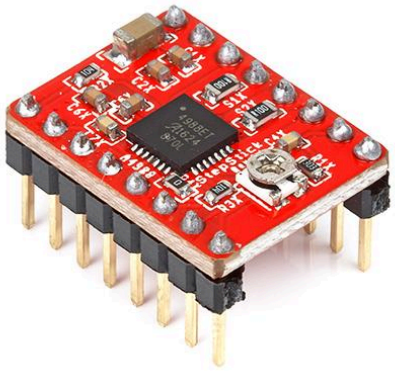


Fig 6.4: A4988

6.5 BUZZER

The buzzer is a sounding device that can convert electric signals into different frequency. It is usually powered by DC voltage. It is widely used in alarms, computers, printers and other electronic products as sound devices. According to different designs and uses, the buzzer can emit various sounds such as music, siren, buzzer, alarm, and electric bell.

Fig 6.5: Buzzer

6.6 16*2 LCD DISPLAY

The LCD screen is an electronic display module and has a wide range of applications. A 16*2 LCD display is a very basic module and is commonly used in various devices and circuits. A 16*2 LCD display means it can display 16 characters per line and there are two such lines. In this LCD each character is displayed in a 5*7 pixel matrix. This LCD has two registers, Command and Data.

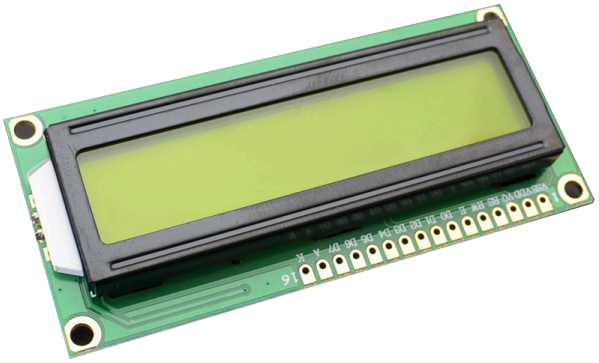


Fig 6.6: 16*2 LCD DISPLAY

6.7 RTC MODULE(DS1307)

The DS1307 serial real-time clock (RTC) is a low-power, full binary-coded decimal (BCD) clock/calendar plus 56 bytes of NV SRAM. Address and data are transferred serially through an I2 C, bidirectional bus. The clock/calendar provides seconds, minutes, hours, days, dates, months, and year information. The end of the month date is automatically adjusted for months with fewer than 31 days, including corrections for leap year. The clock operates in either the 24-hour or 12- hour format with AM/PM indicator. The DS1307 has a built-in power sense circuit that detects power failures and automatically switches to the backup supply. Timekeeping operation continues while the part operates from the backup supply.

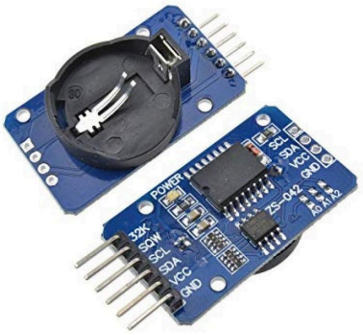


Fig 6.7: RTC MODULE

6.8 PUSH BUTTONS

A push button switch is a mechanical device used to control an electrical circuit in which the operator manually presses a button to actuate an internal switching mechanism. Push button switches rely on a simple in-out actuation mechanism. They can be employed to break (off) or initiate (on) a circuit. Alternatively, they can provide input for the user interface of a piece of equipment or start/stop a particular function.

.

Fig 6.8: PUSH BUTTON

6.9 BEARINGS

6.9.1 RADIAL BEARINGS

Radial bearings are designed specifically to support a radial load. A radial load is a force that occurs perpendicular to a rotating shaft. Radial bearings are ball-based bearings that support a radial load.

Fig 6.9.1: RADIAL BEARING

6.9.2 THRUST BEARINGS

Thrust bearings are designed specifically to support an axial load. An axial load is a force that occurs along the axis of a rotating shaft.

Fig 6.9.2: THRUST BEARING

6.9.3 LINEAR BEARINGS

Linear ball bearings provide linear movement in one direction.

Fig 6.9.3: LINEAR BEARING

6.10 PC COOLING FAN

A PC cooling fan is used to create suction for the vacuum pump.

Fig 6.10: PC COOLING FAN

6.11 LEAD SCREW

The lead screw is attached to the stepper motor via a shaft coupler. It helps in the upward and downward motion of the arm.

Fig 6.11: Lead Screw

6.12 BELT

The belt helps in the 360° rotation of the robotic arm. It connects the base of the arm to the stepper motor.

Fig 6.12: BELT

6.13 RELAY

It is an electrical switch operated by an electromagnet.It has an operating voltage of 250V AC and operating current of 10A.

CHAPTER 7

CIRCUIT DIAGRAM

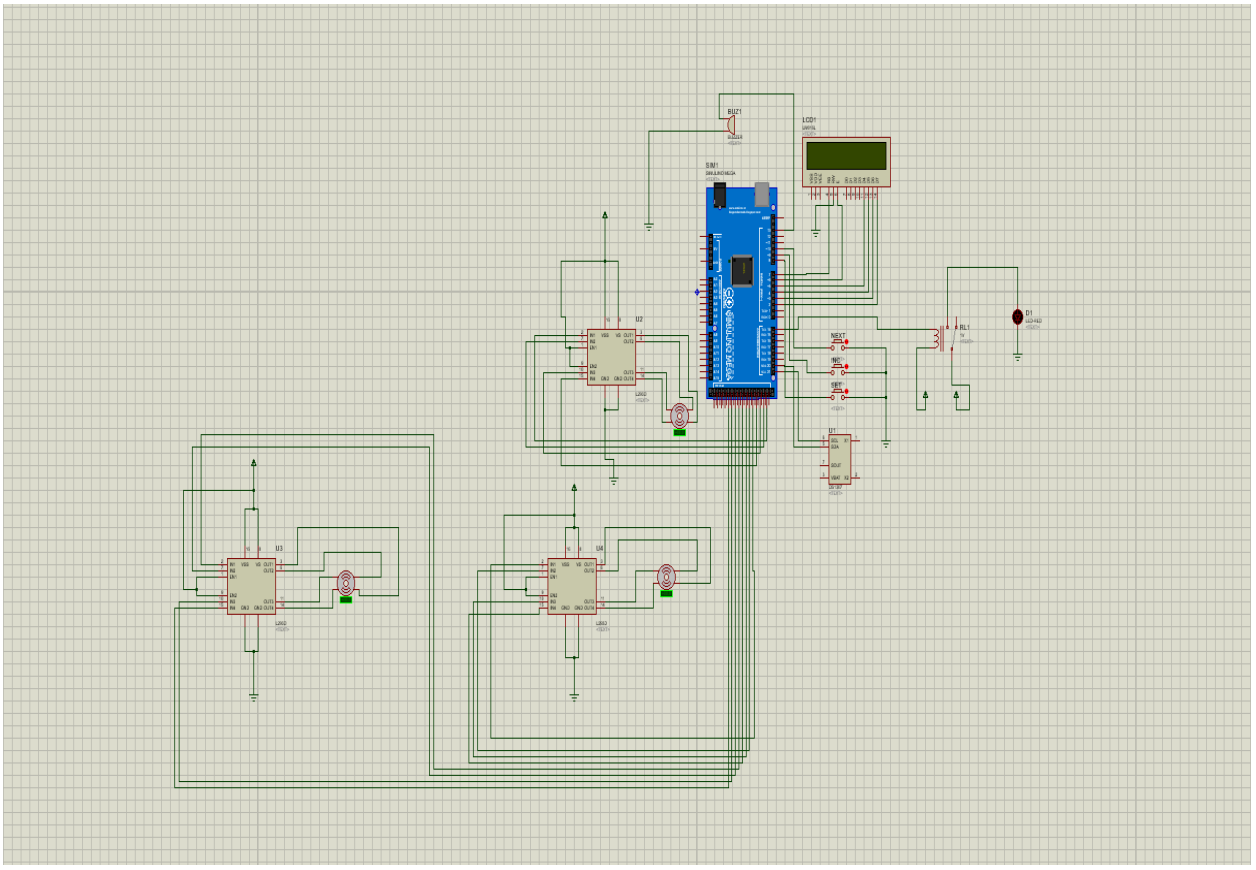


Fig 7.1: Circuit Diagram

CHAPTER 8

SOFTWARE PLAN

Program

["//"-implies comments]

```
//Importing required libraries for interfacing
#include <Wire.h>
#include<EEPROM.h>
#include <RTClib.h>
#include <LiquidCrystal.h>

//Initializing pins with variable names
LiquidCrystal lcd(22, 24, 26, 28, 30, 32);   //initialize LCD
RTC_DS1307 RTC;
int temp,inc,hours1,minut,add=11;
int next=16;
int INC=15;
int set_mad=14;
const int stepPiny = 3; //Y.STEP
const int dirPiny = 6 ; // Y.DIR
const int stepPinx = 2; //X.STEP
const int dirPinx = 5 ; // X.DIR
const int stepPinz=4;   //Z.STEP
const int dirPinz = 7; // Z.DIR
int Contrast=75;
```

```
#define buzzer 34

int HOUR,MINUT,SECOND;

//function to step up various functionality
void setup()
{

 Wire.begin();
 RTC.begin();
 Serial.begin(9600);
 lcd.begin(16,2);

//Defining mode of pins

 pinMode(INC, INPUT);
 pinMode(next, INPUT);
 pinMode(set_med, INPUT);
 pinMode(buzzer, OUTPUT);
 pinMode(34,OUTPUT);
 pinMode(stepPinx,OUTPUT);
 pinMode(dirPinx,OUTPUT);
 pinMode(stepPiny,OUTPUT);
 pinMode(dirPiny,OUTPUT);
 pinMode(stepPinz,OUTPUT);
 pinMode(dirPinz,OUTPUT);
//writing certain pins high for convenient mode of operation
 digitalWrite(set_mad,HIGH);
 digitalWrite(INC,HIGH);
 digitalWrite(next,HIGH);
 digitalWrite(53,Contrast);
```

```
//setting cursor and printing required things on screen
    lcd.setCursor(0,0);
    lcd.print("Medicine alarm");

    lcd.setCursor(0,1);
    lcd.print(" Using Arduino  ");
     delay(2000);
     lcd.setCursor(0,0);
    lcd.print("By ECE STUDENTS  ");
    lcd.setCursor(0,1);
    lcd.print("   GEC THRISSUR   ");
     delay(2000);

 if(!RTC.isrunning())
 {
 RTC.adjust(DateTime(__DATE__,__TIME__));  //setup time of RTC module from time of PC
 }
}

//function that run periodically
void loop()
{
   int temp=0,val=1,temp4;
   DateTime now = RTC.now();
   if(digitalRead(set_mad) == 0)      //set medicine time-1
   {
     lcd.setCursor(0,0);
   lcd.print("  Set Medicine  ");
   lcd.setCursor(0,1);
   lcd.print("  Reminder time ");
    delay(2000);
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("Enter Time 1");
    defualt();
```

```
   time(1);
  delay(1000);
  lcd.clear();
  lcd.setCursor(0,0);      //set medicine time-2
  lcd.print("Enter Time 2");
  defualt();
  delay(1000);
  time(2);
  lcd.clear();
  lcd.setCursor(0,0);       //set medicine time-3
  lcd.print("Enter Time 3");
  defualt();
  time(3);
  lcd.setCursor(0,0);
  lcd.print("Medicin reminder");
  lcd.setCursor(0,1);
  lcd.print("  time has set  ");
  delay(2000);
}
lcd.clear();
lcd.setCursor(0,0);
lcd.print("Time:");
lcd.setCursor(6,0);
lcd.print(HOUR=now.hour(),DEC);
lcd.print(":");
lcd.print(MINUT=now.minute(),DEC);
lcd.print(":");
lcd.print(SECOND=now.second(),DEC);
lcd.setCursor(0,1);
lcd.print("Date: ");
lcd.print(now.day(),DEC);
lcd.print("/");
lcd.print(now.month(),DEC);
lcd.print("/");
```

```
 lcd.print(now.year(),DEC);
 match();     //checking match of time set by user and time on RTC module
 delay(20000);
}

//Function to things seen on screen by default
void defualt()
{
  lcd.setCursor(0,1);
  lcd.print(HOUR);
  lcd.print(":");
  lcd.print(MINUT);
  lcd.print(":");
  lcd.print(SECOND);
}

//Function to set alarm time and feed time into Internal eeprom

void time(int x)
{
  int temp=1,minuts=0,hours=0,seconds=0;
    while(temp==1)
    {
     if(digitalRead(INC)==0)
     {
      HOUR++;
      if(HOUR==24)
      {
       HOUR=0;
      }
      while(digitalRead(INC)==0);
     }
lcd.clear();
      lcd.setCursor(0,0);
```

```
 lcd.print("Enter Time ");
   lcd.print(x);
    lcd.setCursor(0,1);
    lcd.print(HOUR);
    lcd.print(":");
    lcd.print(MINUT);
    lcd.print(":");
    lcd.print(SECOND);
    delay(100);
    if(digitalRead(next)==0)
    {
     hours1=HOUR;
     EEPROM.write(add++,hours1);
    temp=2;
    while(digitalRead(next)==0);
    }
    }
    while(temp==2)
    {
if(digitalRead(INC)==0)
     {
      MINUT++;
      if(MINUT==60)
      {MINUT=0;}
      while(digitalRead(INC)==0);
     }
      lcd.clear();
      lcd.setCursor(0,0);
    lcd.print("Enter Time ");
   lcd.print(x);
    lcd.setCursor(0,1);
    lcd.print(HOUR);
    lcd.print(":");
    lcd.print(MINUT);
```

```
lcd.print(":");
    lcd.print(SECOND);
    delay(100);
      if(digitalRead(next)==0)
      {
       minut=MINUT;
       EEPROM.write(add++, minut);
       temp=0;
       while(digitalRead(next)==0);
      }
    }
    delay(1000);
}




// Function to chack medication time


void match()
{
  int tem[17];
  for(int i=11;i<17;i++)
  {
    tem[i]=EEPROM.read(i);
 }
  if(HOUR == tem[11] && MINUT == tem[12])
  {
   stepperT3();
  delay(500);
   stepperT2_1();
  delay(500);
  stepperT1_1();
  delay(500);
```

```
digitalWrite(17,HIGH);
  delay(1500);

stepperT1_2();
delay(500);
stepperT2_2();
delay(500);

stepperT1_1();
delay(500);

digitalWrite(17,LOW);
  delay(1500);

stepperT1_2();
delay(500);

 lcd.setCursor(0,0);
 lcd.print("  Take Group One  ");
 lcd.setCursor(0,1);
 lcd.print("     Medicine     ");
 beep();
 beep();
 beep();
}

 if(HOUR == tem[13] && MINUT == tem[14])
 {
  stepperT3();
delay(500);
 stepperT2_1();
delay(500);
stepperT1_1();
delay(500);
```

```
   digitalWrite(17,HIGH);
     delay(1500);

   stepperT1_2();
   delay(500);
   stepperT2_2();
   delay(500);

   stepperT1_1();
   delay(500);

   digitalWrite(17,LOW);
     delay(1500);

   stepperT1_2();
   delay(500);

     lcd.setCursor(0,0);
    lcd.print("  Take Group Two  ");
    lcd.setCursor(0,1);
    lcd.print("     Medicine     ");
     beep();
    beep();
    beep();
    beep();
    }

if(HOUR == tem[15] && MINUT == tem[16] )
    {
      stepperT3();
   delay(500);
    stepperT2_1();
   delay(500);
```

```
  stepperT1_1();
  delay(500);

  digitalWrite(17,HIGH);
    delay(1500);

  stepperT1_2();
  delay(500);
  stepperT2_2();
  delay(500);

  stepperT1_1();
  delay(500);
digitalWrite(17,LOW);
    delay(1500);

  stepperT1_2();
  delay(500);

   lcd.setCursor(0,0);
   lcd.print("Take Group Three ");
   lcd.setCursor(0,1);
   lcd.print("     Medicine    ");
   beep();
   beep();
   beep();
   beep();
   }

}
//function to set buzzer beep
void beep()
{
   digitalWrite(buzzer,HIGH);
```

```
 delay(500);
   digitalWrite(buzzer, LOW);
   delay(500);
 }

 //X=T1(Top stepper in scara)
 //Y=T2(Bottom stepper in scara)
 //Z=T3(stepper in medicine box)

 void stepperT1_1(){
  //DOWN
  digitalWrite(dirPinx,HIGH); // Enables the motor to move in a particular direction
 // Makes 200 pulses for making one full cycle rotation
 for(int x = 0; x < 3500    ; x++) {
 digitalWrite(stepPinx,HIGH);
 delayMicroseconds(500);
 digitalWrite(stepPinx,LOW);
 delayMicroseconds(500);
 }
 delay(1000); // One second delay

  }
void stepperT1_2(){
  //UP
  digitalWrite(dirPinx,LOW); // Enables the motor to move in a particular direction
 // Makes 200 pulses for making one full cycle rotation
 for(int x = 0; x < 3500    ; x++) {
 digitalWrite(stepPinx,HIGH);
 delayMicroseconds(500);
 digitalWrite(stepPinx,LOW);
 delayMicroseconds(500);
 }
```

```
delay(1000); // One second delay



 }

 void stepperT2_1(){

   //Anti clockwise
   digitalWrite(dirPiny,LOW); // Enables the motor to move in a particular direction
// Makes 200 pulses for making one full cycle rotation
for(int x = 0; x < 1000    ; x++) {
digitalWrite(stepPiny,HIGH);
delayMicroseconds(500);
digitalWrite(stepPiny,LOW);
delayMicroseconds(500);
}
delay(1000); // One second delay



 }

 void stepperT2_2(){

   // clockwise
   digitalWrite(dirPiny,HIGH); // Enables the motor to move in a particular direction
// Makes 200 pulses for making one full cycle rotation
for(int x = 0; x < 1000    ; x++) {
digitalWrite(stepPiny,HIGH);
delayMicroseconds(500);
digitalWrite(stepPiny,LOW);
delayMicroseconds(500);
}
```

```
delay(1000); // One second delay



 }

 void stepperT3(){
   //Anti clockwise
   digitalWrite(dirPinz,LOW); // Enables the motor to move in a particular direction
// Makes 200 pulses for making one full cycle rotation
for(int x = 0; x < 35    ; x++) {
digitalWrite(stepPinz,HIGH);
delayMicroseconds(500);
digitalWrite(stepPinz,LOW);
delayMicroseconds(500);
}
delay(1000); // One second delay



 }
```

CHAPTER 9

CONCLUSION

The goal of our project is to provide a healthy and tension-free life to those users who are regularly taking pills. Our Smart Medicine Box reminds us to take medicines regularly. And the trouble of remembering the names of medicine is not required. Our project would make the life of elderly people easy and they won't have to depend on someone else for their medications.

CHAPTER 10

FUTURE SCOPE

An SMS alert system can also be incorporated into our project, which gives warning messages to the caretaker. The SMS system can be implemented with the help of sensors. An emergency button can also be included, which gives alarms to Doctors and caretakers in times of crisis.

We can also provide a rechargeable power supply to make it a better product.

With the help of a mobile application, we can also provide support to monitor medicine in the container.

REFERENCES

- International Journal of Biomedical Engineering and Science (IJBES), Vol. 5, No. 3/4, October 2018 DOI: 10.5121/ijbes.2018.5401 1 **DESIGN OF A SAFE AND SMART MEDICINE BOX Roy ABI ZEID DAOU1,2, Khalil KARAM1 , Hiba ZEIDAN1, Ali HAYEK3 and Josef BORCSOK3**
- International Journal of Biomedical Engineering and Science (IJBES), Vol. 5, No. 3/4, October 2018 DOI: 10.5121/ijbes.2018.5401 1 **DESIGN OF A SAFE AND SMART MEDICINE BOX Roy ABI ZEID DAOU1,2, Khalil KARAM1 , Hiba ZEIDAN1, Ali HAYEK3 and Josef BORCSOK3**
- **A Smart Pill Box with Remind and Consumption Confirmation Functions** ,2015 IEEE 4th Global Conference on Consumer Electronics (GCCE) **Huai-Kuei Wu1,Member, IEEE, Chi-Ming Wong2, Pang-Hsing Liu1 , Sheng-Po Peng1, Xun-Cong Wang1 , Chih-Hi Lin1 and Kuan-Hui Tu**
- **Smart Medicine Box System** 2018 IEEE MULTIDISCIPLINARY CONFERENCE ON ENGINEERING TECHNOLOGY by **Hiba Zeidan,Khali karam,Roy Abi Zeid Daou,Ali Hayek,Jos Boercsoek,**
- **Smart Medicine Reminder Box** by **Sanjay Bhati Harshid Soni ,Vijayrajsinh Zala Parth Vyas** IJSTE - International Journal of Science Technology & Engineering | Volume 3 | Issue 10 |
- Buzzer datasheet
- A4988 datasheet
- NEMA 17 stepper motor datasheet
- https://youtu.be/1QHJksTrk8s
- https://youtu.be/IM3d0dlAOJk
- DS1307 datasheet
- 16*2 LCD display datasheet