Design of Smart Home Pharmacy

Project Team:

Bayan Drabee   **Fatima Sbeih**

Supervisor:

Eng. Ali Amro

Submitted to the College of Engineering

in partial fulfillment of the requirements for the degree of

Bachelor degree in Biomedical Engineering

Palestine Polytechnic University

December , 2017

I

الإهداء

بسم الله الرحمن الرحيم

(يَرْفَعِ اللَّهُ الَّذِينَ آمَنُوا مِنكُمْ وَالَّذِينَ أُوتُوا الْعِلْمَ دَرَجَاتٍ وَاللَّهُ بِمَا تَعْمَلُونَ خَبِيرٌ)
صدق الله العظيم

لَكَ الْحَمْدُ يَارَبَّنَا حَمْدًا دائماً طاهراً طَيِّبًا مُبَارَكًا فيه.....

إلى سيدنا وحبيبنا وشفيعنا رسول الإنسانية والسلام والإسلام محمد بن عبد الله عليه أفضل الصلاة الوسلام

ما أجمل الأحلام نحياها......وظلك يا أبي......عبر السنين......وأنت تسقينا الحنين......سيظل قلبك طيباً يا والدي

مثل النخيل

والدي العزيز

والى من ساندتني يوم ضعفي ....الى حبيبتي التي شاركتني همي وحزني.... البمن ذرفت الدموع من أجلي....
الى من سقتني الحب في صغري حتى أرتوت منه عروقجسدي....
أمي الحبيبة

إلى الروح التي سكنت روحي زوجي الغالي

لكم يا كل الصامدين في خنادقكم .... يا أبطال المعاناة خلف القضبان...........الأسرى والمعتقلين

إلى كل المناضلين والمقاومين الذين يعيدون كتابة التاريخ بأحرف من دماء...............إلى شهداءفلسطين

إلى الذين مهدوا لنا طريق العلم والمعرفة......... إلى جميع أساتذتنا الأفاضل...........إلى كل عاشق للعلم .... وكل
محب للتقدم .... وبراما ملموسا كما يرى الشمس في وضع النهار.......

الى من احتضنتني كل هذا الكم من السنين .....فلسطين الحبيبة

Acknowledgment

We would like to thank Palestine Polytechnic University, College of Engineering, and

Electrical Engineering Department. Thanks from our hearts for all support and for this worthy learning

environment.

Also thanks to the head of hopefulness, our parents.

We would like to thank everybody shared in success of this work either by suggestion,

directives, or tips. Thanks to Eng. Ali Amro for their great efforts in supervision, suggestion,

and providing experience to accomplish this work. Also we want to thank both Eng. Fida'aAlja'fra

and to Dr. Ramziqawasmeh For all helps and worthy suggestions and tips to accomplish this project.

Abstract

Aged patients needs special care. Usually they take several types of medicines in different times. This requires help from healthy people to those aged patients to keep the medicines in safe place, and give each medicine in time.

The idea of this project is to design smart pharmacy that can provides the aged patients with all the facilities that can help them without any need from healthy people.

Since heat, air, light, and moisture may damage the medicine, the smart pharmacy consists of chambers to keep medicines from air, and sensors that measure the temperature, and humidity of the chambers, to store the medicine properly and keep it from getting damaged.storage conditions by measuring its temperature and humidity, and automatically activating the fan when they are out of range.The user can specified the number of times per day the dose should be taken and the time for each dose according to the doctor prescription, then the system sends sms message to patient's mobile phone to remind him to take each drug in the specified time. Furthermore, the system gives the user an indication of the remaining drug pieces in each chamber and the expired date for each.

The main problem which faces the patients especially the aged one taking numerousness medicines which may cause oblivion to the aged patients. This project in this thesis is about controlled circuit that designed to organize the time of drags via using special technique ,such as engines, micro controller ,and all of the electronic element to the patients who need many drags.

الملخص

يحتاج المرضى المسنين رعاية خاصة عادة ما يأخذون عدة أنواع من الأدوية في أوقات مختلفة وهذا يتطلب مساعدة من الأشخاص الأصحاء للمرضى المسنين للحفاظ على الأدوية في مكان امن وإعطاء كل دواء في الوقت المناسب.

فكرة هذا المشروع هو تصميم الصيدلية الذكية التي يمكن أن توفر للمرضى المسنين مع جميع المرافق التي يمكن أن تساعدهم دون أي حاجة من الأشخاص الأصحاء.

وبما أن الحرارة والهواء والضوء والرطوبة قد تتلف الدواء تتكون الصيدلية الذكية من غرف للحفاظ على الأدوية من الهواء وأجهزة الاستشعار التي تقيس درجة ألحرارة والرطوبة من ألغرف لتخزين الدواء بشكل صحيح ومنعها من التلف.

تحتوي الصيدلية الذكية على نظام برودة ينشط تلقائيا عند زيادة درجة حرارة ألغرفة وعلاوة على ذلك سيتم تنشيط نظام إنذار في وقت أخذ كل دواء وعندما ينتهي عدد حبات الدواء.

الصيدلية لديها القدرة على إرسال رسالة إلى المستخدم باسم الدواءو كمية المتبقية وتاريخ انتهاء الصلاحية,المشكلة الرئيسية التي تواجه المرضى وخاصة المسنين أخذ الأدوية العديدة التي قد تسبب النسيان للمرضى المسنين,هذا المشروع باستخدام تقنية خاصة مثل ألمحركات وحدة تحكم ألجزئيوجميع العناصر الإلكترونية تساعد المرضى .

Table of Contents

Chapter One: - Thesis Overview

Chapter two : - Drugs

Chapter Three: - Technology Background

4.6List of Content

List of Figures

NO	*Figure*	Page NO.
3.1	DHT11 Temperature & Humidity Sensor	11
3.5	Fan Model Type: EC-4010xxxx	12
3.6	Real Time Clock (RTC)	13
3.7	Keypad	14
3.8	Servo Motor	15
3.9	LDC Crystal	16
3.10	GSM Shield	17
3.11	Arduino MEGA **2560**	20
4.1	Block diagram of Smart Home Pharmacy	24
4.2	keypad and Arduino connection	25
4.3	DHT11 and Arduino connected	26

List of Tables

Chapter one

Thesis Overview

1.1 **Project Idea Description.**

1.2 **Project Motivation.**

1.3 **Project Aims.**

1.4   **Literature Review and Related Work.**

1.5 **List of Abbreviation.**

1.6 **Economical Study.**

1.7 **Schedule Time.**

1.1 Project Idea Description

The proposed project aims to design a portable smart pharmacy consists of several chambers, each chamber for specific type of medicine and digitally locked and requires entering password to be opened.

The pharmacy has several features such as keeping the medicine at FDA standard storage conditions by measuring its temperature and humidity, and automatically activating the fan when they are out of range.

The user can specified the number of times per day the dose should be taken and the time for each dose according to the doctor prescription, then the system sends sms message to patient's mobile phone to remind him to take each drug in the specified time. Furthermore, the system gives the user an indication of the remaining drug pieces in each chamber and the expired date for each.

.

1.2 Project Motivation

Medicines are meant to help make patients feel better. They also can help treat a medical condition. But if medicines aren't taken exactly the right way and time, they can be harmful. According to the Centers for Disease Control and Prevention (CDC), more than 60,000 young children [1]go to the emergency room each year because they got into medicines while their parents or caregivers were not looking. That's why it's so important to prevent the children from reaching the medicines. These side effects create the idea of this project to design a digital safe smart pharmacy that can keep all medicines out of child's reach and sight, and provide the user with counting and timing system that can count each medicine pieces remain in each chamber.

1.3 Project Aims

- The project to alert the medication on time or according to the doctor's prescription by timing system and Alert the patient in the event of a lack of medication via RGB-LED.
- keeping the pharmacy locked from reach of child and enabling the user to it by inserting password to the keypad also Enter the name and expiration date of the medicine.
- The pharmacy is portable so it will be available whenever the patient needs it
- Keeping the medicine at temperature (10-27) c, by controlling the temperature inside the pharmacy and running the fan when it is greater than 27c.
- Send a message to the patient's phone to Remember taking the drug, The mobile phone history shows: the name of the medicine, the doctor's prescription, the amount of the remaining and the expiry date.
- Alert the patient in case of lack of medication via RGB-LED.
- Show the time and date permanently, the name and the room of medicine when time of medicine occure by LCD-screen.

1.4 Literature Review and Related Work

The project study comes to accomplish many previous studies that include different techniques for the smart pharmacy, the previous design is about "smart pharmacy card"[2] which automates the patient's medical history and medical review so that each new description can be analyzed by drug makers Tudermin Ivet conflicts with diseases, with prescription and non-prescription drugs The card is documented, and the model that will appear will use the integrated circuit card ("smart card") and erasable data. [3]. There were many graduation projects for the smart pharmacy, and this is one of the projects used The pharmacy provides a product delivery service for patients, also products for all agegroups and all needs facilitate the choice and

availability of the product or not. Through that IPHON mobile application we deal with people especially in ABHA cityin this application[4].

This project has been designed to investigate and select the optimal methods to suit and meet the requirements of the drug by adding the time alert to avoid the expiration date of the drug by inserting the date of the medication with the carbide and the pharmacy locked by using servo motor. It alert the patient of medicine time or in case of medicine's decreasing by sending a message on the patient's mobile.

1.5 List of Abbreviation

The following list includes all of the abbreviation scripted in the project

Table 1.1: List of Abbreviation

Abbreviation	Full word
FDA	**Food and Drug Administration**
LCD	**Liquid Crystal Display**
RTC	**Real Time Clock**
AC	**Alternating Current**
DC	**Direct current**
PWM	**Pulse Width Modulation**
AM	**After Midnight**
PM	**Past Midnight**
GSM	**Global System for Mobile Communication**
C	**Celsius**
UV	**Ultraviolet**
RAM	**Random Access Memory**
SIM	**Subscriber Identity Module**
GND	**Ground**

1.6   Economical Study

The total project cost is shown in the table

Table 1.2:Component Cost.

no	Type	price	Quantity
1	Arduino Mega	80 JD	1
2	DHT11	60 JD	1
5	Fan	20 JD	1
6	RTC	30 JD	1
7	Keypad	5 JD	1
8	Servomotor	15 JD	5
9	GSM	50 JD	1
10	LED	0.5 JD	18
11	LCD	10 JD	1
	Total	300JD	

1.7 Schedule Time

In this time plan for project tasks is due to the time zone for each of the 1.7 Schedule Time shown in the table 1.4 and table 1.5.

Table 1.4 Schedule Time For First Semester

Task \ Week	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Collect Information	■	■	■	■											
Basic Design					■	■	■	■							
Specification Design									■	■	■	■			
Documentation									■	■	■		■	■	■
Advance feature											■	■	■	■	■

Table 1.5 Schedule Time For the Second Semester

Task \ Week	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Full Designing	■	■	■	■											
Purchasing the components				■	■	■	■	■							
System Implementation						■	■	■	■	■	■	■			
System analysis									■	■	■	■			
Documentation											■	■	■	■	

Chapter two

Drugs

2.1 Types of drugs

2.2 The reasons for maintaining the drug

2.3 Methods of keeping drugs

2.4 Risks of drugs

Introduction

Drug is any substance used in the prevention, diagnosis, alleviation, treatment, or cure of disease or a chemical compounds or natural contain effective in treating the disease article, in addition to other materials such as materials to give the medication color attractive, or preservatives, or material gain medicine smell and taste is acceptable[5] .

2.Types of Drugs :

Drugs are Classified according to drugs source , exchange method ,Drug Effect, The method of using it.

According to Source drug ,drugs that rely on chemical substances (metals), or plants, or any other medical sources, to treatment of pathological Status. According to

According to exchange method drugs are classified to drugs without prescription: the medicines that are available in front of the person who visits the pharmacy, and do not cause serious side effects, including: some types of pain relievers, toothpaste, drugs with prescription: drugs that depends pharmacist in dispensing, the existence of a prescription with the person , which depend from doctor, Examples include: medicines that contain substances that affect the work of the nervous system[6].

According to drug Effect drugs are classified to quick drugs effect: drugs that show results in a short period of use, up to a maximum of one day ,Slow drugs effect: drugs that require several days to show results on the patient.

According to method of using the drug: classifies this type of drugs into two categories, namely drug-use oral : drugs that are taken by mouth, such as tablets, capsules, drugs used injection: drugs that are given to the patient by (injection), in the area of artery blood , considered to be of a quick effect, compared with drugs that are given by mouth[7] .

2.2 Importance of maintaining the drug

As Keep heat, light and humidity can cause the active drug in the biodegradable components, and its necessary to keep the drugs in healthy place , Preventing the child poison incidents often that result from taking the drugs[8].

2.3 Methods of keeping drugs:

Storage play an important role in keep all of these types of drugs until the end of the expiry date, but proper storage of drugs need to be familiar with the way some things that store drugs[9].

Medicines should be kept out of reach the children , drugs are stored in a dry place away from heat and humidity , Kept medicines in their packaging , in traveling Drug store in medicine bags ,do not keep drugs in the bathroom or near the kitchen or in other wet places because they cause damage of medicines ,do not save liquid medicines in the freezing temperature inside the freezer ,do not store multiple drugs in the same packaging ,do not remove the labels from the containers because they include the end of the important instructions and how to store the medicine[10].

In order to complete treatment, patient should Check the name of the medicine before taking it Check the times of taking medication ,Inquired about the side effects of drugs ,Read the drug name and dose before take ,be sure to take medicine at a fixed date , Be sure of the drug expires and keep drug in specific place[11] .

2.4 Risks of drugs

Taking wrong drug may cause several side effect the child is unable to express what it is suffering from a disease , Weakening their immune system, increases the acidity of the stomach can lead to ulcers in the intestine ,kill beneficial bacteria to the human body , cause stomach infections, bottom the esophagus infections ,and may leads to kidney failure in the long term[12].

Chapter Three

Technology Background

3.1Sensors.

3.2 Cooling system.

3.3 Timing Clock.

3.4Security System.

3.5 LCD Screen.

3.6 GSM Transmitter.

3.7 Microprocessor (Arduino).

This chapter provides technical and theoretical background about all system elements and components ,required to implement the project successfully

3.1 Sensors

Several types of sensors are required in the project such as temperature sensor , optical sensor , in this section more details about each sensor are studied to choose the best one for each stage .

3.1.1 Humidity-Temperature Sensor

Heat is one of the most important types of energy and the temperature is measured in Relative humidity is defined as the .three global units Kelvin, Celsius and Fahrenheit ratio between the actual water vapor pressure of a sample of air and the pressure of saturation vapor at a given temperature. Relative humidity is measured by R.H%.

This sensor is to measure the temperature and humidity inside the pharmacy rooms in order to store it in good condition. The temperature must be within the range (15-25)C° If the temperature of the pharmacy exceeds 25 C°, this sensor will affect the sensitivity of the temperature increase and thus sends a signal to the Arduino Which operates the cooling system to reduce the temperature, and works to measure the humidity inside the pharmacy no more than 60% relative humidity in normal storage conditions, in case the humidity increased than 60% The sensor works to send a signal to Arduino to operate a system that reduces moisture.

The following table shows a comparison between different Humidity-Temperature Sensor.

Table 3.1 Comparison Between Humidity-Temperature Sensor

Features	RHT03	DHT11
Response time	6 s	2 s
Sensitivity Humidity	0.1% RH	0.5% RH
Sensitivity Temperature	0.1℃	+-1 ℃
Accuracy Humidity	±2% RH	± 5% RH
accuracy Temperature	±0.5℃	±2℃
Measurement Range Humidity	0-100% RH	20–90% RH
Measurement Range Temperature	-40 – 80 ℃	0-50℃

The Sensor That will be used is DHT11, because it is fast and high accuracy, and because its output is a digital signal, also on the economic side its more affordable and available in the local market[13].
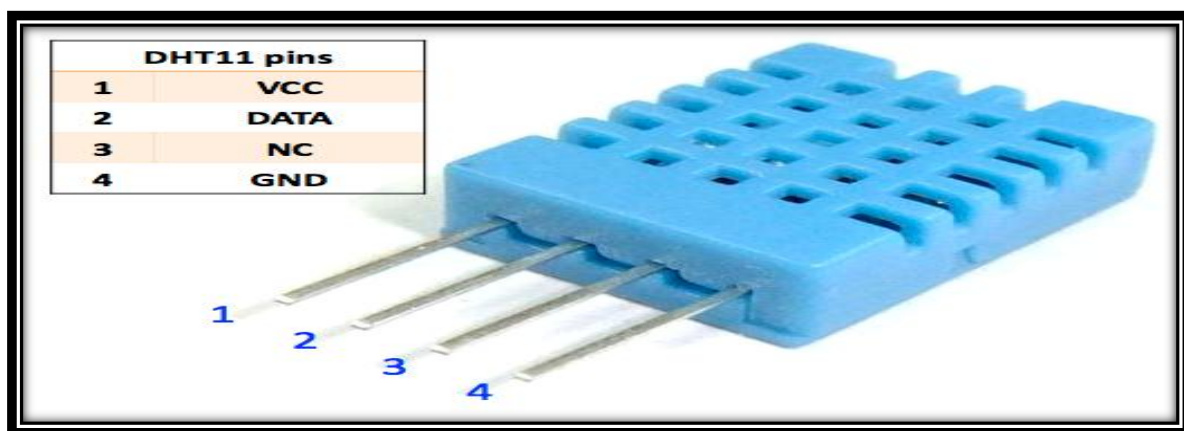

Fig 3.1 DHT11 Temperature & Humidity Sensor

3.2   Cooling system

A cooling system must be operated in order to store the medicine in good condition. This system is an air cooling fan inside the pharmacy. The fan is an electric device used for ventilation in the hot air to provide a comfortable atmosphere.

In its simplest form, the fan consists of an electric motor connected to the end of blades. When the engine rotates, the feathers generate an antenna current.

The fan operates on the basis of generating a difference in pressure and thus displacing the air from the higher pressure point towards the lower pressure resulting from the movement of the blades. This operation is done by fan blades (slanting) that displace air touching the blade surface as they rotate. Because the air is removed from one side of the blade surface, the pressure rises slightly on one surface and creates a partial vacuum on the other surface of the blade. Because pressure (air here) is equal, it drives a quantity of air away from the airless surface towards it to fill this vacuum. This process continues with fan rotation[14].

The fans produce a large quantity of air current but at low pressure, we will use the fan Model Type: EC-4010xxxx, It has several features >17DBA Noise, Life Hour50000, Fan Speed 6000RPM, Voltage 5VDC and Dimension :40 x 40 x 10 mm.
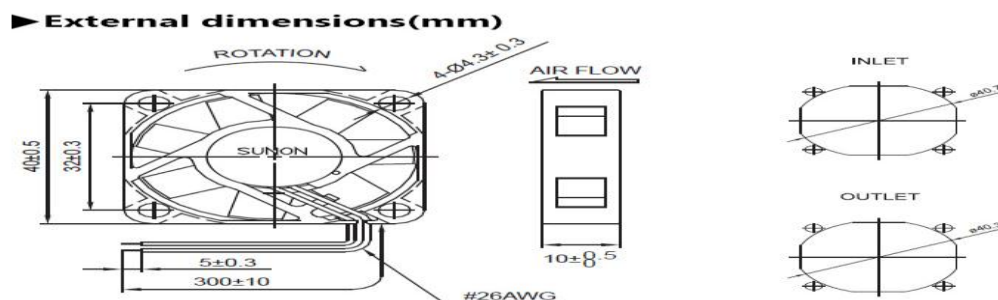
Fig 3.5 Fan Model Type: EC-4010xxxx

3.3 Timer Controller

To gain an accurate timing for the medicine expiration date, there is something required : (a timekeeping chip). The DS1302 trickle-charge timekeeping chip contains a real-time clock/calendar and 31 bytes of static RAM. RTC is found in several electronic devices that need to save time. It is connected with a micro-controller via a simple serial interface of pins. The 4 pins will be connected to another digital pins of the micro-controller. The real-time clock is used to reduce the consumption ; this is known as the most accurate calendar that provides informations per seconds, minutes, hours, day, date, month, and year . At the end of the month, an automatically adjusted happens in less than 31 days, and that happens for months including corrections for leap year. The clock works in either the 24-hour or 12-hour format with an AM/PM indicator .RTCs often have an alternative power source, so users can continue saving time when the main power source is off or unavailable. This alternative power source then is normally a lithium battery[15]
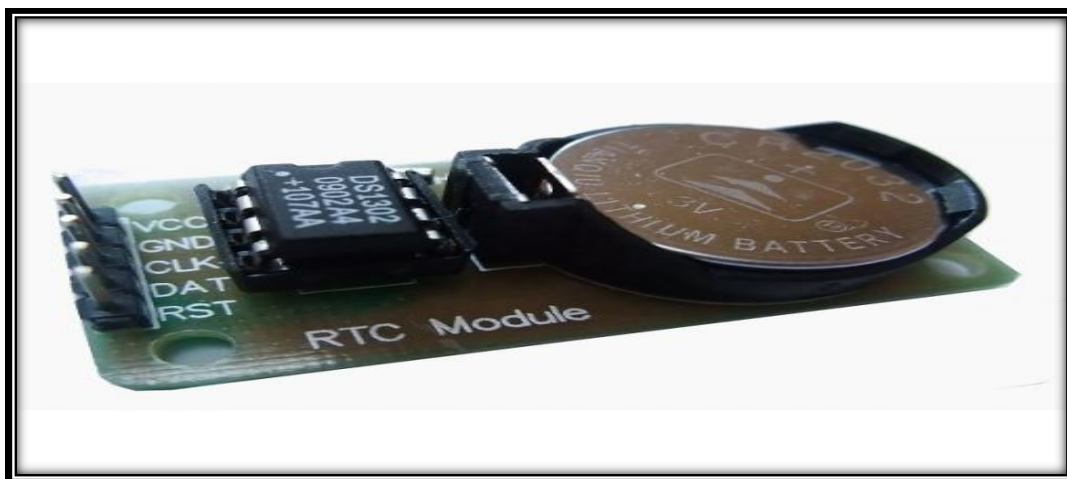
Fig 3.6 Real Time Clock (RTC)

3.4   Security System

3.4.1 Keypad

The Smart pharmacy will be secured by using 4 pins with its numbers in order to make it hardly reachable to the children. that pin is needed when the expiration date of the medicine is occurred, and when you want to open the medicine or to refill it once agian or when you want to throw it away. The system is connected with a servo locker using matrix style keyboard with the Arduino which is controlled. A keyboard is a set of buttons that arranged in a panel; that panel includes numbers, digits, symbols or alphabetical letters. It consists of a matrix of rows and columns that connect its rows to be a feeding source and its columns as a part- feeder route[16] .

The button may be functioned as a switch, and if you press on it, it allows

the current to pass through.

The 7 pins will be connected to the digital pins of the micro-controller

-         4-1are the rows

-         5-7 are the columns
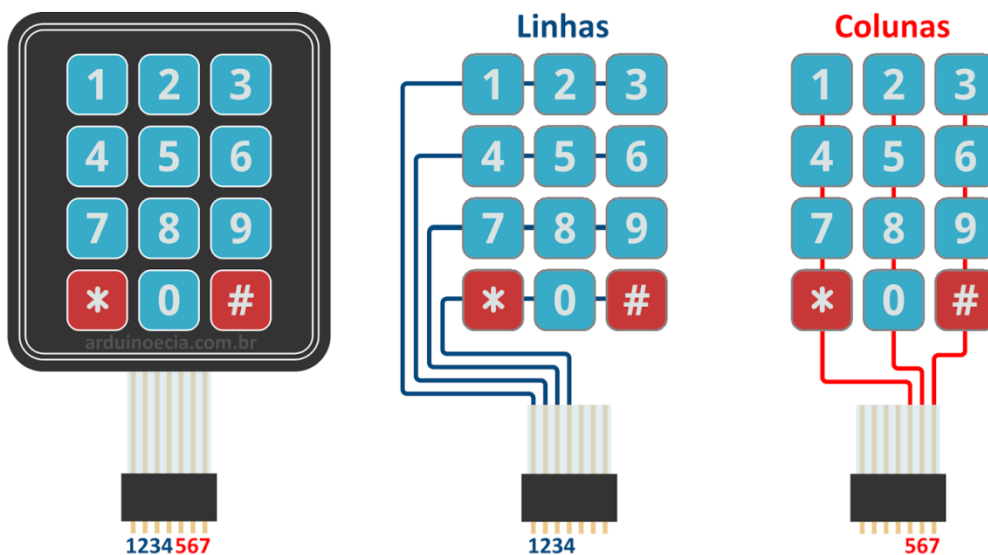
Fig 3.7 Keypad

3.4.2 Servo Motor

A servomotor is a rotary actuator or linear actuator that allows for precise control of angular or linear position, velocity and acceleration . always rotates at full speed. This type of servomotor is a basic form of simple and cheap devices used for models controlled by Arduino ,It consists of a suitable motor coupled to a sensor for position feedback. Stander Servo Motor, It's an engine that is able to rotate from 0 to 180 in both directions with clockwise and counterclockwise. Also it works with DC motors. Standard Servo Motor has many distinctive features: Firstly the linear S relationship between voltage and speed. Secondly It's specialized in developing the motor shaft which we call it the potentiometer. The Potentiometer is a variable resistance that leads to change the outside voltage. Due to the value of the voltage appears, the electronic circuit determines the position of the motor shaft.. It also requires a relatively sophisticated controller, often a dedicated module designed specifically for use with servomotors[17].

When the signal pin of the motor enters the Analog PWM Pin set Arduino will drive the servo motor to rotate 180 degrees to open the mechanical fan such as the lock.
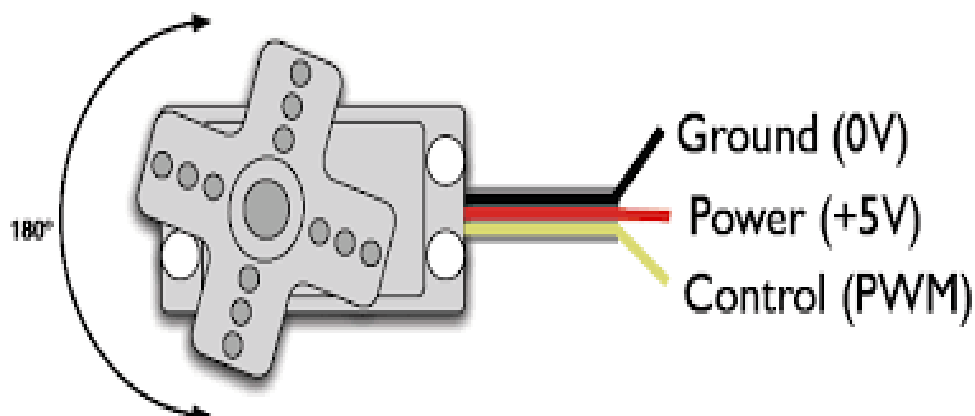
Fig 3.8 Servo Motor

3.5   LCD Screen

Smart Pharmacy will show the date, clock, prescription and doctor's prescription for Crystal Library's LSD library allowing us to control the LED screens compatible with the Hitachi HD44780 driver. There are many of them out there, and we will use the 16-pin interface. It is connected with Arduino through pin number[18]. LED has a parallel interface, which means that the microcontroller has to handle multiple pins and interface simultaneously to control the screen. The interface consists of the following pins:

Select between the controls in the LEED memory for which you are writing data. You can choose either the data log, which holds what goes on the screen, or the help log, where the LED controller is looking for instructions on what to do next. Read / Write (R / W) pin which selects read mode or write mode, Enable pin which lets write to records ,8 pins data (D0-D7). The cases of these pins (high or low) are the bits you write to a record when you write, or the values you read when reading them.

The Hitachi-compatible LCD monitors can be controlled in two modes: 4-bit or 8-bit. The 4-bit mode requires seven I / O pins from Arduino, while 8-bit mode requires 11 pins. To display the text on the screen, you can do most of the thing in 4-bit mode.

Fig 3.9LDC Crystal

3.6   GSM Transmitter

The Arduino GSM shield allows the Arduino board to connect to the Internet, send and receive SMS, and make voice calls using the GSM library. Among other things, GSM supports incoming and outgoing voice calls, and data communications[19].

To access a network, it must be shared with a mobile phone operator, a device compatible with a receiver such as a shield or a mobile phone, and a subscriber ID card (SIM) card. The network operator provides a SIM card that contains information such as a mobile number, and can store limited amounts of contacts and SMS. SIM card operates on a network that can be used within the surrounding area. Although the GSM connection is identical with the Arduino only the 2, 3, 7 and GND digital pins are used to connect between Arduino and the modem.

SIM900a will use shield implant provides a way to use sniffer cell phone network to receive data from remote location. This shield allows this to be achieved through short message service methods.

Fig 3.10GSM Shield

3.7 Controller (Arduino)

Microprocessor is a very important part in our system and it is the brain of the system that connect and control all sensors, achieve security for the system and sending SMS messages. Arduino controller is used for simplicity to deal with it and compatibility with other electronics , its supplied with 5 v minimum voltage via external battery or by directly connecting by USB cable which enables the user to upload the code and also to supply it with the required 5 v to fully enable its features[20] .

There are many types of Arduino microcontroller Arduino UNO and Arduino MEGA. the Arduino MEGA 2560 is designed for more complex projects and it more suitable for our project because it has more digital and analog pins that give the best coverage of the pharmacy if more than one sensor is needed to measure temperature in the smart pharmacy's chambers .

Pins mapping for the Arduino MEGA by 0-53 digital pins and 2-13 PWM A0 – A15 Analog pins .
The Uno board can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the GND and Vin pin headers of the POWER connector.

The board can operate on an external supply from 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may become unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

➢ Vin. The input voltage to the Uno 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[22].

➢ 5V.This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V),or the VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board. We don't advise it.

➢ 3V3. A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50mA.

➢ GND. Ground pins.

Each of the 54 digital pins on the Mega can be used as an input or output, using pin Mode, digital Write, and digital Read () functions. They operate at 5 volts. Each pin can provide or receive 20 mA as recommended operating condition and has an internal pull-up resistor (disconnected by default) of 20-50k ohm. A maximum of 40mA is the value that must not be exceeded on any I/O pin to avoid permanent damage to the microcontroller.

In addition, some pins have specialized functions:

➢ Serial: 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.

➢ External Interrupts: 2, 3, 18, 19, 20and 21 . 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[23].

➢ PWM: 2-13 . Provide 8-bit PWM output with the analog Write () function.

➢ SPI: 50 (SS), 51 (MOSI), 52 (MISO), 53 (SCK). These pins support SPI communication using the SPI library.

➢ LED: 13. There is a built-in LED driven by digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off[24].
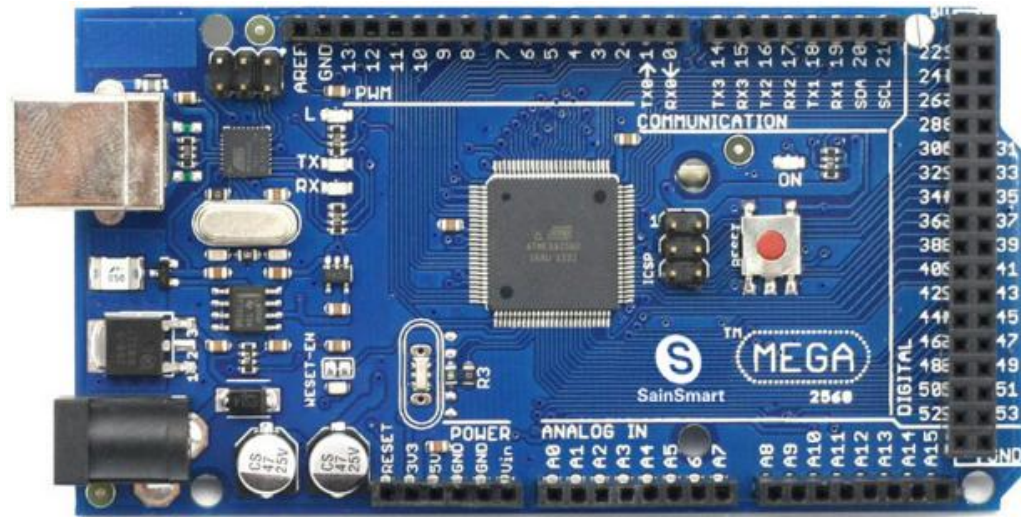
Fig 3.11

Arduino MEGA **2560**

Chapter Four

Design and Implementation of the Pharmacy circuit

4.1 Design principle.

4.2Keypad.

4.3Temperature and humidity sensor (DHT11).

4.4Time control.

4.5 GSM Transmitter.

4.6Display screen.

4.7Motor.

4.8Cooler.

4.9 Leds.

4.10 Buzzer.

4.11Controller.

4.1 Design principle

 In this chapter first standard and recommended requirementof smart home pharmacy circuit represented then based on the requirements, design details and block diagram represented ,and finally , component selection criteria and brief explanation of the selected components for each stage stated .

The system was supposed to store the medicine in appropriate temperature ,it work as a reminder for the elders and for the handicapped persons to remind them about the time and the doctor prescription of the medicine when the time occurs as a protection to make it far from child hand a keypad was added to lock the pharmacy from reach of child , the system shown the expiration date ,the system sent an SMS message to the users mobile phone showing the following information medicine name expiration date.In order to fulfill the preceding tasks ,block diagram of the system shown in figure (4.1) is designed .

```
                          ┌──────────┐
                          │   Fan    │
                          └──────────┘
                               ↑
┌──────────────┐         ┌──────────┐         ┌──────────────┐
│ Keypad   4X4 │────────▶│          │────────▶│   Display    │
└──────────────┘         │          │         └──────────────┘
┌──────────────┐         │          │         ┌──────────────┐
│ Temperature  │         │          │────────▶│ Servo Motor  │
│  Humidity    │────────▶│Controller│         └──────────────┘
│ (sensor HD11)│         │          │         ┌──────────────┐
└──────────────┘         │          │────────▶│     GSM      │
┌──────────────┐         │          │         └──────────────┘
│ TimControl   │         │          │         ┌──────────────┐
│              │────────▶│          │────────▶│    Leds      │
│    RTC       │         └──────────┘         └──────────────┘
└──────────────┘              ↓
                          ┌──────────┐
                          │  Buzzer  │
                          └──────────┘
```
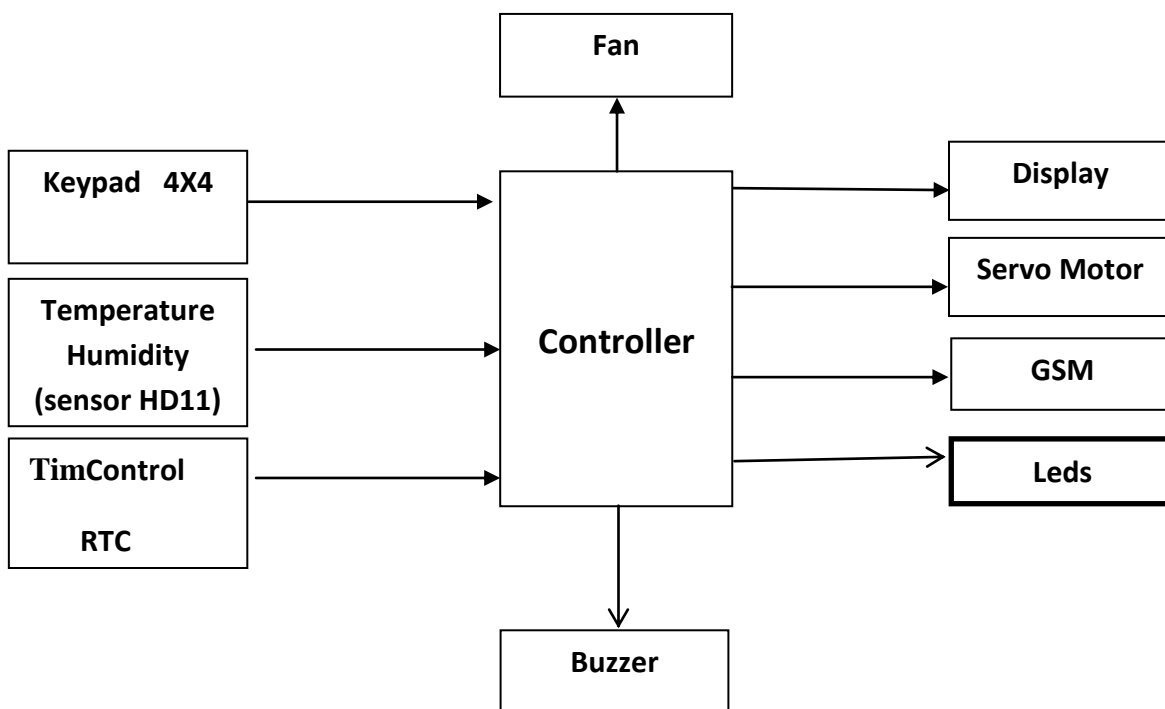
4.2Keypad

Keypad was required to enter the password ,required to open the pharmacy , to enter the expire date of each drug .It connected to the microcontroller as shown in figure (4.2)So thatkeypad was connected directly to adriano and need 5 volt to work arduino , VI=5 VOLT from arduino,Rows R1, R2, R3 and R4 , Columns C1, C2, C3 and C4,keypad pins(1,2,3,4,5,6,7,8) arduino digital pins(32,33,34,35,36,37,38,39).
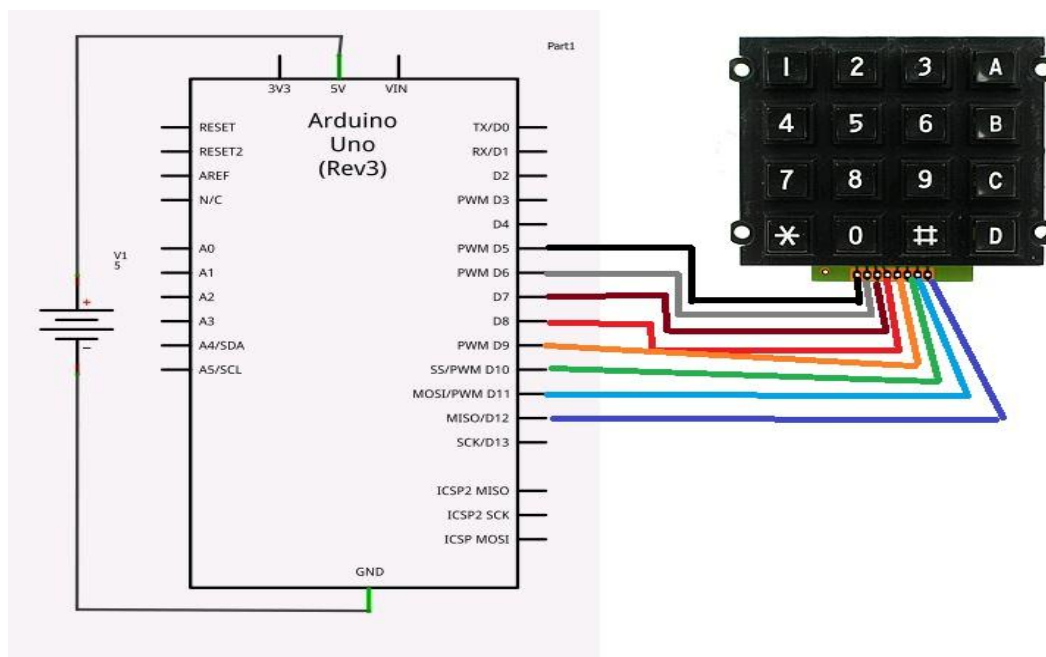
Fig. 4.2 keypad andArduino connection

4.3Temperatureand humidity sensor (DHT11)

In this project DHT11 sensor it used to sense the temperature and humidity and if it above or below standard rangetempreture (25C) it provided the arduino to control ,temperature & humidity sensing technology, it ensures high reliability and excellent long-term stability, offering excellent quality,fast response Time (Seconds)) 6 S , Its small size, low power consumption ,Measurement

Range(0°C- 50),DHT11's power supply need 5V DC ,VIN =5 VOLT from arduino ,it connected to arduino and need 10k ohms to save from high current.

see the data sheet in picture .
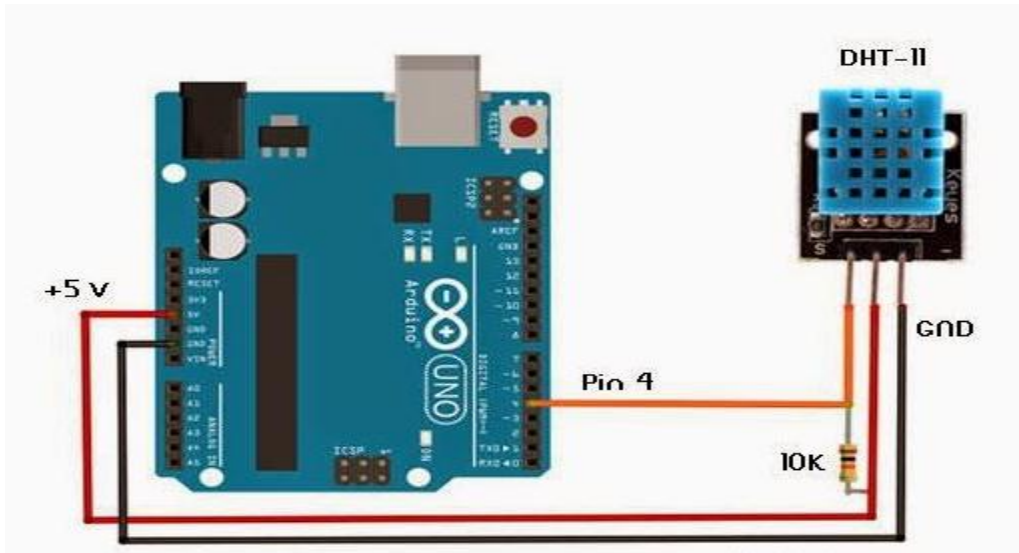
Fig. 4.3 DHT11 and Arduino connected

4.4Time control

In this project real time clock used as timerthrough limiting the time required to give the drug when it reached that time given the arduino signal to control, It communicated with a microprocessor via a simple serial interface ,SCL (clock SDA (data)Through them as much as reading data , need 5 voltage to work and connected directlyVIN =5 VOLT , RTC pins (5V ,GND, SCL , SDA) arduino analog pins (21,20).

as shown the datasheet in the picture.
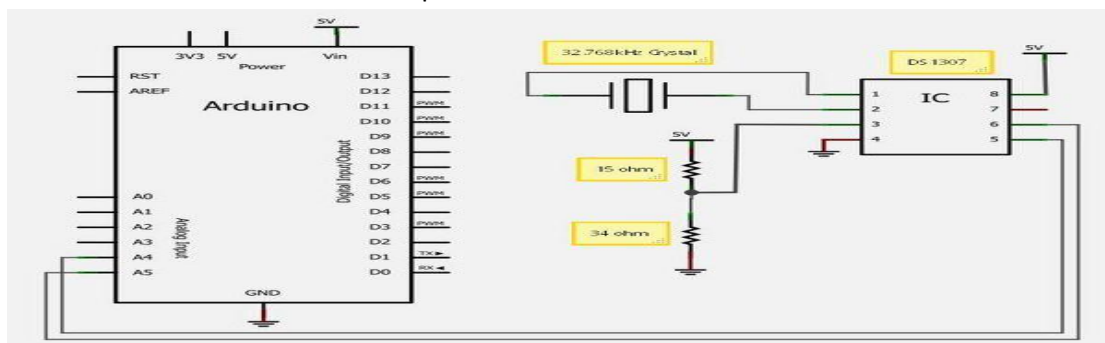
Fig. 4.4 RTC connected to Arduino

4.5 GSM Transmitter

It was used to sent massage to the user such as (expire date , name of drug and time to take it),Operating voltage 5V supplied from the arduino board,VIN =5 VOLTfrom arduinoconnected with arduino on pins 2, 3 (Software Serial) because pins 2,3 are digital pins and pin 7 (reset),sent messages by programming the text of the message inside the code and for the specified time the program was sent the message.

figure(4.5) depicts the datasheet.
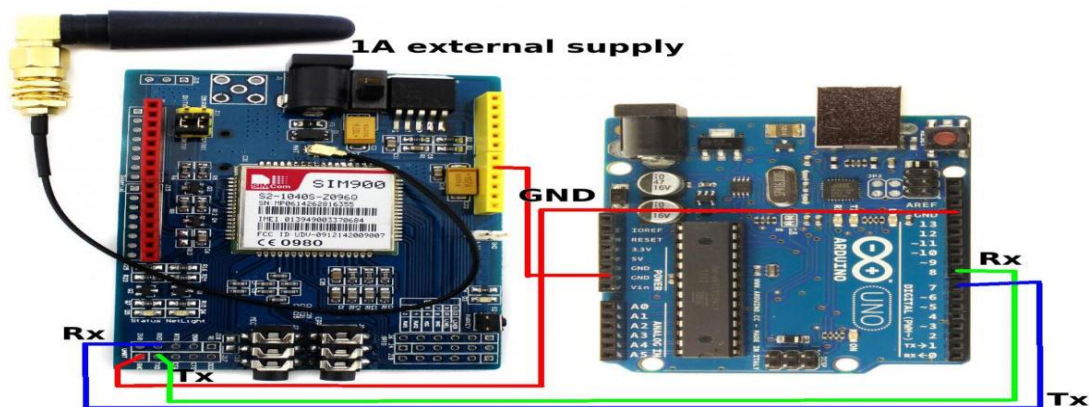
Fig. 4.5 GSM Transmitter and Arduino connection

4.6 Display screen

In this project LCD Screen used to display time ,date , expire date and name of drug , it need 5 volt to workVIN =5 VOLT ,it was connected directly to arduino , LCD pins (Rsenable ,D4,D5,D6,D7)arduinopins (40,41,42,43,44,45)respectively A register select (RS) pin that controls where in the LCD's memory you're written date.
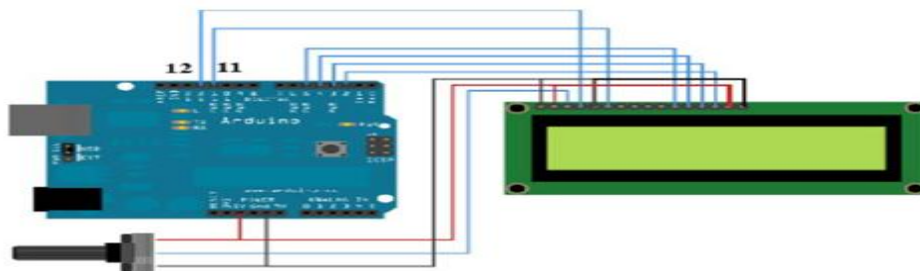
Fig 4.6LCDconnected to Arduino

27

4.7 Servo Motormodel – SG90 -9g

 In this project Servo Motorwasopened the room when the medicine is time taken need one power line, one ground, and one control pin the servo motor has a connector with three pins directly with Arduino, The darkest or even black one is usually the groundconnected this to the Arduino GND ,Connected the power cable that in all standard should be red to 5V on the ArduinoConnect the remaining line on the servo connector to a digital pin on the Arduino , servo analog pins (8,9,10) 5volt signal ground ,figure(4.7) depicts the datasheet.
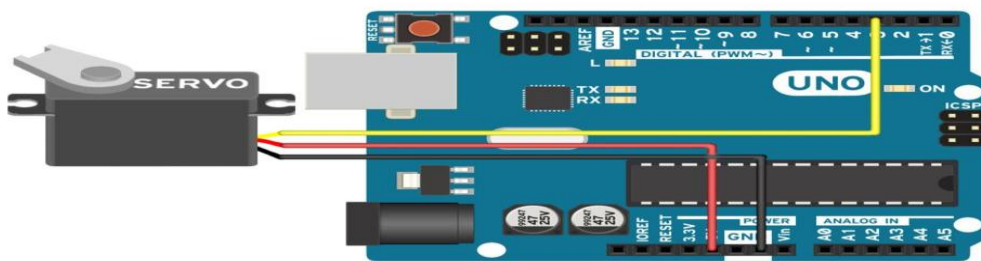
Fig. 4.7servo motor and Arduino connection

4.8Fan

When the temperature is high, the arduino send signal to fan to work ,used to cool the pharmacy to save the medicine ,diode   The diode to protect against currents generated by the existence of a current file in a magnetic field. This magnetic field has a reduced energy so that this energy returned to the transistor or circuit until it is dipped with a diode protection   , resistor protect for high current, it need 9 volt to work It is connected to an external power supply.
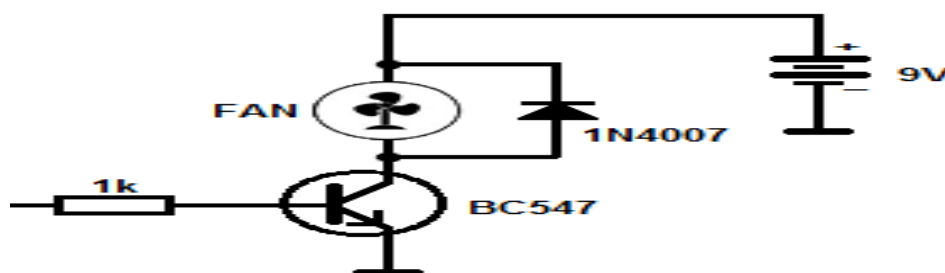
Fig 4.8circuit of Fan with arduino

4.9 Buzzer

In this project Buzzer gave sound (name of drugs)when it time , it used when the user enter error password it gave sound ,and it gave another sound true password ,) So that Buzzerwas connected   directly to adriano and need 5 volt to work arduino , VI=5 VOLT.

Fig. 4.9 Buzzerand Arduino connection

4.10 leds

In this project leds on when the pharmacy open and another leds explained how many medicine in the room ,So that ledswere connected   directly to adriano and need 5 volt to work arduino , VI=5 VOLTfrom arduino.

Fig. 4.10 Ledand Arduino connection

4.11 Controller

Microprocessor (Arduino) used to process data, where it received data from the keypad and received the temperature and humidity of the DHT11 and then processed ,after that sent massage to GSM , sent to LCD the expire date of the medicine .

A microcontroller (also microcomputer, MCU or μC) is a small computer in a single

integrated circuit consisting internally of a relatively simple CPU, clock, timers, I/O

 ports, and memory. Microcontrollers are designed for small or dedicated applications

thus, in contrast to microprocessors used in personal computers, 5Vregulated power supply used to power the microcontroller and other components on the board.

Chapter 5

System Implementation and Testing

5.1   Implementation

5.2 Testing

5.2.1 Unit Testing

5.2.2 Integration Testing

5.2.3 System Testing

5.2.4 Acceptance Testing

The idea of the project came from seeing the patients who suffer from chronic diseases such as pressure , diabetes and other diseases, where our system provides the patient with easy handling of taking the drug in it's time by the doctor supervisor, who is not much interested in timing or the amount of medicine remaining or the expire date.   Here comes the importance of electronic pharmacy, in the fact that the patients has full control over these matters.

5.1 Implementation

We began to gather information about the implementation of the project by asking some patients and the importance of providing such a device for them. They were very interested for the idea and encouraged it. First by thinking about the design phase of the device, we started how to link dates with the real time of occurrence. Here we have the use of (RTC 1307) because of the time synchronization of the drug with real time. To allow the patient to determine the intend time to use keypad. Then to display these values to the patient we resorted to the use of the LCD screen. If the patient cannot move there is no other problems, we decided to use GSM to send a text message to the patient's guardian. To determine the number of pills remaining in the box, RGB LED was used. Here comes the role of microprocessor to connect all these devices with each other and for many of us we have used Arduino MEGA for this purpose.

The prototype system is applied and tested in this chapter. Figure 5.1 depicts the pharmacy containing solution pathways, sensors, RTC,servo motor,Keypad , led and the Arduino controller is implemented.

Fig 5.1 the pharmacy containing the electronic pieces inside it

An implementation of each stage in the system is performed according to the system requirements. The hardware and software components arethen examined by doing the appropriate test for each stage as will be discussed in the following sections.

5.2   Testing

Testing plan is one of the modules of the system analysis and thedesign, which has greater importance. After system is ready it shouldgo for testing. Testing is one of the critical phases.Various Testing Methods are[4] :

5.2.1 Unit Testing

In Unit Testing,individual components are tested for their validations to ensure that they operate correctly[5].

5.2.1.1 Test Temperature and Humidity With Arduino

The DHT11 Sensory has been linked with Arduino with the path of the solution. Because of the test, the sensore measures when raising the temperature and humidity of the pharmacy, results are obtained on the screen that was connected with the sensore and the Arduino through the Arduino programming as shown in Figure 5.2.

Fig 5.2temperature and humidity and display.

When the temperature and pharmaceutical humidity are raised, the sensor is affected quickly and the results appear on the screen that is connected with the sensor and the arduino as shown in Figure 5.3

Fig5.3 temperature and humidity

5.2.1.2 Test **Real Time ClockWithArduino**

The time and date were determined by the RTC, which was programmed with Arduino. The tests were conducted and showed the exact time and date on which the full work was performed to perform the tasks required of the pharmacy. Thus, through time knowing, Patient will be alret and recieved a message on Phone

5.2.1.3 **TestRGB-LED Testing WithArduino**

RGB LED means red, blue and green LEDs. These products were used as a medicine pill counter when the patient took the medicine to give the patient a warning according to Table 5.1 in order to attract the attention of the patient so as to provide the medication at the appropriate time.

Table 5.1:RGB-LED Reading

Drug pills	RGB-LED Reading
10-8	green
7-4	blue
3-0	red

Figure 5.5 shows the place of RGB-LED for the drawer and here it is green, as the number of pills inside the drawer is 8-10.
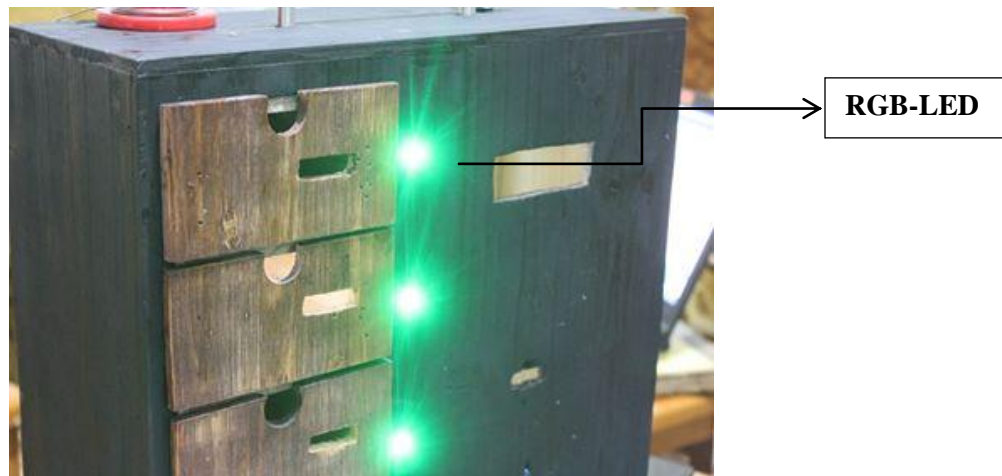
Fig5.4 RGB-LED

5.2.2   Integration Testing

The next level is called Integration Testing. In this many unit testedmodules are combined into subsystems, which are then tested. Thegoal here is to see if the modules can be integrated properly. Thistesting activity can be considered testing the design.

5.2.2.1   Temperature and Humidity Sensorwith The Fan Test

As mentioned previously, the temperature and humidity of the pharmacy is measured through the Sensor Dht11 and because the pharmacy is working to save the medicine when increasing the temperature of 27 and the humidity of 60% Arduino gives the fan to run to reduce the temperature and humidity and return to normal condition without affecting the drug

and so the drug is saved in the pharmacyas shown in Table 5.2.

Table 5.2: Reading of humidity-temp Sensor with LCD &FAN.

Sensor red	LCD Reading	FAN
tem	25>	ON
hum	60%>	ON

Figure 5.6 shows the connection of the sensor with the Arduino and shows the temperature on the LCD ,the fan is turned on if the temperature and humidity increase.
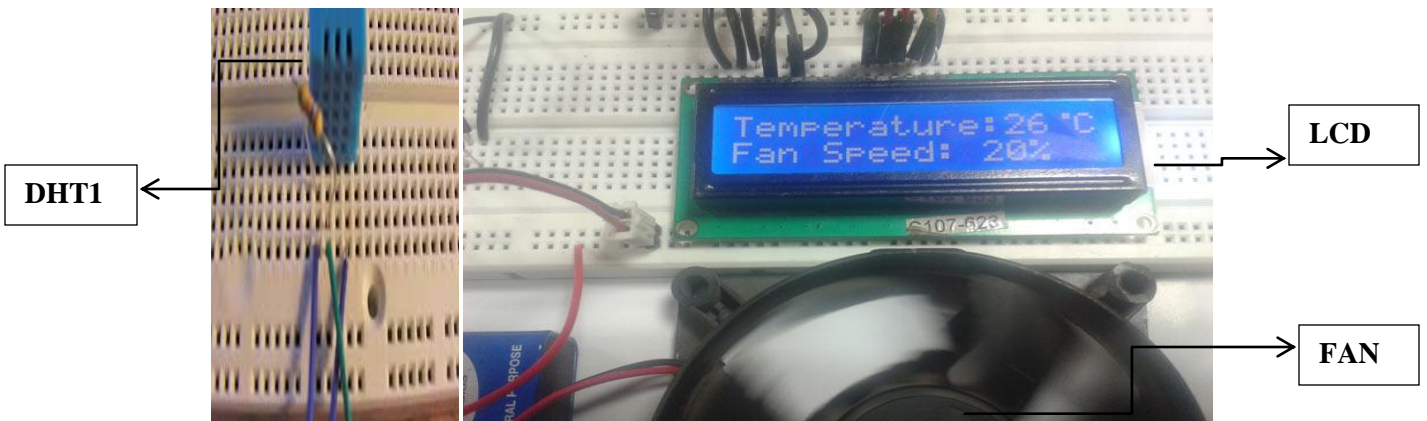
Fig5.6 temperature and fan sensors and display

5.2.2.2 Real Time test with the Keypad, Servo Motor and GSM

When the time of the drug occurs the pharmacy is working on warning the patient with a voice and optical signal and which is determined from the piece RTC and send a message to the patient's phone is done by a piece of GSM works to store the patient number and send the message to him through a piece inside and also at on the time of medicine Servo Motor opens the lock of the tractor for the patient to takeas in the form 5.7.

Fig5.7 Real Time with the Keypad, Servo Motor and Sim900

5.2.3 System Testing

After testing each piece separately and connecting two or more pieces and testing them, we worked to connect all parts of the project so that the work coincides with each other to give the desired results while the date of the drug occurs, it is turned on the light and it works as an alert as well as the sound alarm and also works as a motor vehicle to unlock, at the same time It sends a message to the patient's phone via a GSM segment and acts as a protective system to enter the hemorrhoid through the keyboard that appears on the LCD screen.

We also used the RBG-LED indicator to count the medicine beads and so was connected to all parts of the electronic project.Figure 5.8 shows the link of the project pieces.
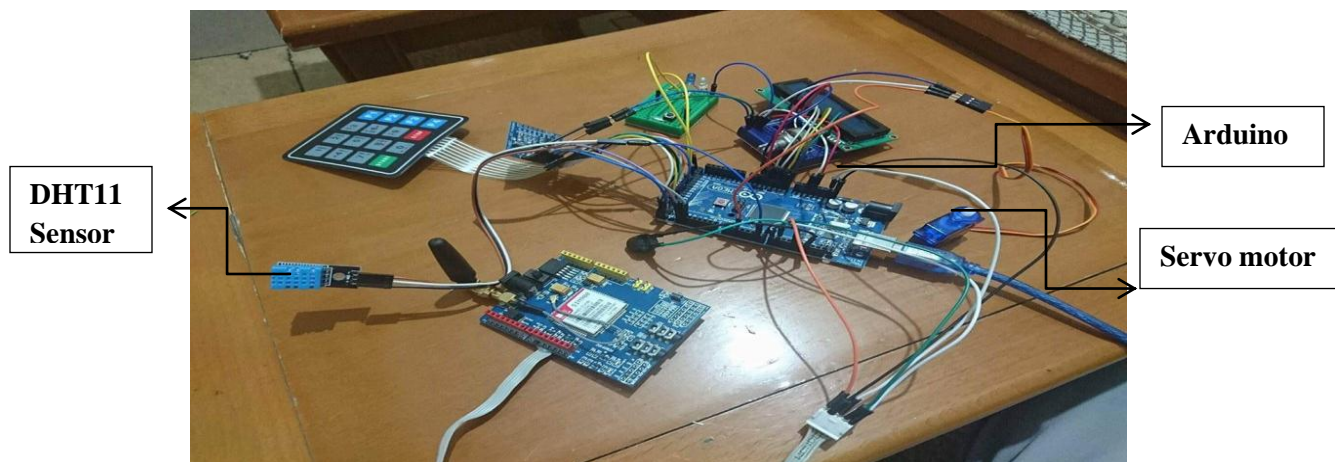
Fig 5.8 all parts of the project

5.2.4   Acceptance Testing

Finally, the electronic parts were connected until the system was completed, the aluminum parts were placed in the wooden box which keeps the electronic parts and the medicine so that the fan was placed on the back side until it works on the cooling and placed the RGB-LED beside the drawer until it alert the patient to the end of the number of pills per drawer. We put the keyboard in the right side of the drawer under the LCD until we enter the required information and LCD is used to show and insert the required data from the patient.

The electronic pieces were placed on the right side of the pharmacy from the inside so that we protect them as shows the Figure 5.9.

Fig5.9 Accepting electronic parts in wood Box

Chapter 6

Software of the Pharmacy circuit

6.1 Flow chart.

In this chapter first standard and software requirement of smart home pharmacy circuit .
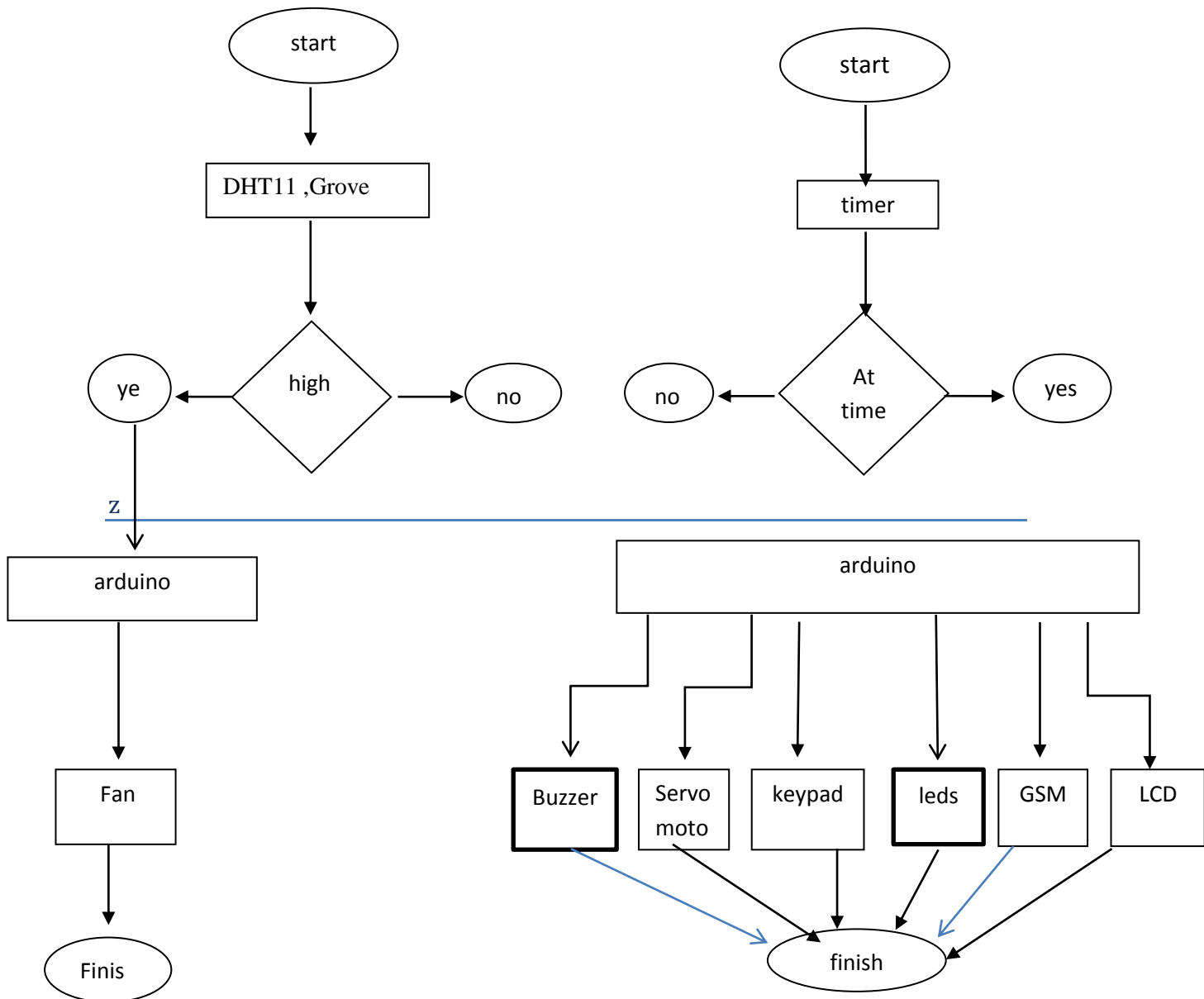
6.2 Software principle.

/* ----- C Program for Arduino based Alarm Clock ---- */

// https://circuitdigest.com/microcontroller-projects/arduino-alarm-clock

/**/

#include<EEPROM.h>

#include <Wire.h>

#include <RTClib.h>

int add_A=11, add_B=55 , add_C=80, add_D=100; // EEPROM ADDRESSEE

String time_HA ,time_MA,time_HB ,time_MB,time_HC ,time_MC,time_HD ,time_MD;

int HOUR,MINUT,SECOND;

int HA,MA,HB,MB,HC,MC,HD,MD;

int temp,inc;

int hours1_A,minut_A,hours1_B,minut_B, hours1_C,minut_C,hours1_D,minut_D;

boolean alarm_A_Activated = false;

boolean alarm_B_Activated = false;

boolean alarm_C_Activated = false;

boolean alarm_D_Activated = false;

//boolean passChanged = false;

int screenOffMsg =0;

char whichKey;

boolean activated = false;

char keypressed;

/**/

#include <LiquidCrystal.h>

```
LiquidCrystal lcd(A0, A1, A2, A3, A4,A5);

RTC_DS1307 RTC;

#define buzzer 12

/************************************************/

 #include <Keypad.h>

 const byte ROWS = 4; //four rows

const byte COLS = 4; //four columns

//define the cymbols on the buttons of the keypads

char keyMap[ROWS][COLS] =

{

  {'1','2','3','A'},

  {'4','5','6','B'},

  {'7','8','9','C'},

  {'*','0','#','D'}

};

byte rowPins[ROWS] = {22, 23, 24, 25}; //Row pinouts of the keypad

byte colPins[COLS] = {26, 27, 28, 29}; //Column pinouts of the keypad

Keypad myKeypad = Keypad( makeKeymap(keyMap), rowPins, colPins, ROWS,
COLS);

boolean AlarmChanged = false;

int redLED =51; // Invalid Key detection

/************************************************/

//define the DHTXX Temperature & Humidity Sensor

#include <dht.h>

dht DHT;

int fan= 53;

#define DHT11_PIN 50
```

```
// Complete project details

//at http://randomnerdtutorials.com

#include <SoftwareSerial.h>

SoftwareSerial SIM900(7, 8); // Configure software serial port

/**************************************************/

 #include <Servo.h>

Servo servo_A ,servo_B ,servo_C, servo_D;  // create servo object to control a servo

// twelve servo objects can be created on most boards

int pos = 0;    // variable to store the servo position

/**************************************************/

// Room LED

#define room_A 32

#define room_B 33

#define room_C 34

#define room_D 35

/**************************************************/

String tempPassword;

boolean Password_Acti = false;

String password="1234";

String medPassword ="7896";

String massage, massage_A, massage_B, massage_C, massage_D;

/**************************************************/

//Medicine Counter Variable

int Count_A =10;

int Count_B =10;

int Count_C =10;
```

```
int Count_D =10;
/***********************************************/
//RGB lED PIN'S Distribution
 int R_A =36 ,G_A =37 ,B_A = 38;
 int R_B =39 ,G_B =40 ,B_B = 41;
 int R_C =42 ,G_C =43 ,B_C = 44;
 int R_D =45 ,G_D =46 ,B_D = 47;


void setup()
{
  Wire.begin();
  RTC.begin();
  lcd.begin(16,2);
  pinMode(buzzer, OUTPUT);
  pinMode(redLED, OUTPUT);
  pinMode(fan, OUTPUT);
/**********************/
 // define pin's (32 -35) as OutPut for Room Lighting
  for (int R =32;R<=35;R++)
  {
   pinMode(R, OUTPUT);
  }
/**********************/
 // define pin's (32 -35) as OutPut for RGB lED PIN'S
  for (int R =36;R<=44;R++)
  {
   pinMode(R, OUTPUT);
```

```
  }
```

/**********************/

```
  tone(buzzer, 2000, 300);

  lcd.setCursor(0, 0);

  lcd.print("    Welcome");

  delay(3000);

  tone(buzzer, 2000, 100);

  lcd.setCursor(0,0);

  lcd.print("Design of smart");

  lcd.setCursor(0,1);

  lcd.print(" home pharmacy");

  delay(3000);

  tone(buzzer, 2000, 100);

  lcd.setCursor(0,0);

  lcd.print(" Bayan drabey");

  lcd.setCursor(0,1);

  lcd.print("Fatima Yasser");

  delay(3000);

  tone(buzzer, 2000, 100);

  lcd.setCursor(0,0);

  lcd.print(" Supervisor :");

  lcd.setCursor(0,1);

  lcd.print("Mr. Ali Amro");

  delay(2500);
```

/**********************/

```
  // Automatically turn on the shield

  // Connect D9 on the shield to the D9 Arduino pin
```

```
  digitalWrite(9, HIGH);

  delay(1000);

  digitalWrite(9, LOW);

  delay(5000);

  // Arduino communicates with SIM900 GSM shield at a baud rate of 19200

  // Make sure that corresponds to the baud rate of your module

  SIM900.begin(19200);

  // Give time to your GSM shield log on to network

  delay(2000);
/***********************/

 if(!RTC.isrunning())

  {

   RTC.adjust(DateTime(__DATE__,__TIME__));

  }
/**********************/

 // Attaches the servo on pin'S (2-5)to the servo object

 servo_A.attach(2);

 servo_B.attach(3);

 servo_C.attach(4);

 servo_D.attach(5);
/**********************/

}
/***************************************/


void loop()

{
```

```
   int temp=0,val=1,temp4; // New parameter

  DateTime now = RTC.now();

/******** to check if Temp. > 27 ********/

int chk = DHT.read11(DHT11_PIN);

  if( DHT.temperature >= 27)

    {

     digitalWrite (fan,HIGH);

    }

  if (DHT.temperature < 27)

    {

     digitalWrite (fan,LOW);

    }

/******** Invalid Key! ********/

char whichKey = myKeypad.getKey();

if( whichKey == '0' || whichKey == '1' || whichKey == '2' || whichKey == '3' ||

        whichKey == '4' || whichKey == '5' || whichKey == '6' || whichKey == '7' ||

        whichKey == '8' || whichKey == '9' ||whichKey == '#' /*|| whichKey == '*'*/ )

 {

   tone(buzzer, 300, 100);

   lcd.clear();

   lcd.setCursor(0, 0);

   lcd.print("  Invalid Key!");

   digitalWrite(redLED, HIGH);

   delay(800);

   digitalWrite(redLED, LOW);

   lcd.clear();
```

```
  }
/******** AAAAA ********/
 if (whichKey =='A')
 {
    tone(buzzer, 2000, 100);
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print(" Set Alarm Time ");
    defualt();     // is a function to display time on LCD
    delay(1300);
    /*****************/
       lcd.clear();
       lcd.setCursor(0,0);
       lcd.print("A -T.Medicine #1");
       lcd.setCursor(0,1);
       lcd.print("B -T.Medicine #2");
       delay (2000);
       lcd.clear();
       lcd.setCursor(0,0);
       lcd.print("C -T.Medicine #3");
       lcd.setCursor(0,1);
       lcd.print("D -T.Medicine #4");
       delay (1500);
    /****************/
    Time_medicine ();   // function to set alarm time
    delay(500);
    lcd.clear();
```

```
  lcd.setCursor(0,0);

  lcd.print("  Alarm time ");

  lcd.setCursor(0,1);

  lcd.print(" has been set  ");

  delay(2000);

 }
/******** BBBBB ********/

if ( whichKey =='B')

 {

   tone(buzzer, 2000, 100);

   lcd.clear();

   lcd.setCursor(0,0);

   lcd.print("    Medicine");

   lcd.setCursor(0,1);

   lcd.print("  Name and Date");

   delay(2500);

   medicine();

 }
/******** CCCCC ********/

if (whichKey =='C')

 {

   tone(buzzer, 2000, 100);

   bool Hum =true;

   int Temp =1;

   while (Hum)

    {

       lcd.clear();
```

```
lcd.setCursor(0,0);

lcd.print("Temperature and ");

lcd.setCursor(2,1);

lcd.print("and Humidity ");

delay(130);


char  set_mad = myKeypad.getKey();

if ( set_mad =='*')

 {

  tone(buzzer, 2000, 100);

  while (Temp ==1)

   {

    Temp_humidity ();

    delay(10);

    char reset = myKeypad.getKey();

    if (reset == '#')

     {

       tone(buzzer, 2000, 100);

       Temp =0;

       Hum =false;

     }

   }

 }

}
/******** DDDDD ********/

if ( whichKey =='D')
```

```cpp
  {
     tone(buzzer, 2000, 100);

   lcd.clear();

   lcd.setCursor(0,0);

   lcd.print("  Time Alarm ");

   lcd.setCursor(0,1);

   lcd.print("  for Medicine");

   delay(2500);

   Display();

  }
/******** passworld ********/

 if ( whichKey =='*')

 {

 tone(buzzer, 2000, 100);

 delay(1500);

 enterPassword();

 }
/******** Display of Time all the Time ********/

 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);
```

53

```
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_A();

match_B();

match_C();

match_D();

counter_A (Count_A);

counter_B (Count_B);

counter_C (Count_C);

counter_D (Count_D);

delay(300);


}




/************************??? Function ???************************/

/****************************************************************/

// functon to display time on LCD

void defualt()

{

lcd.setCursor(3,1);

lcd.print(HOUR);
```

```
  lcd.print(":");

  lcd.print(MINUT);

  lcd.print(":");

  lcd.print(SECOND);

  delay (1300);

}

/***********************************************/

/* function to buzzer indication */

void beep()

{

  tone(buzzer, 2000, 500);// Send 1KHz sound signal

  delay(100);

  tone(buzzer, 2000, 500); // Send 1KHz sound signal

  delay(100);

}

/***********************************************/

/*Function to set alarm time and feed time into Internal EEPROM*/

void time_A(int H,int M )

{

  if(H >=24)

    {

     H=0;

    }

  if(M >=60)

    {

     M=0;

    }
```

```
lcd.clear();

lcd.setCursor(0,0);

lcd.print("Set Alarm Time ");

lcd.setCursor(0,1);

lcd.print(H);

lcd.print(":");

lcd.print(M);

lcd.print(":");

lcd.print(00);

delay(10);


hours1_A=H;

 EEPROM.write(add_A++,hours1_A);

minut_A=M;

EEPROM.write(add_A++, minut_A);

temp=0;

delay(2000);


}
/*****************************/

void time_B(int H,int M )

{

  if(H >=24)

    {

    H=0;

    }

  if(M >=60)
```

```
      {
       M=0;
      }
      lcd.clear();
      lcd.setCursor(0,0);
      lcd.print("Set Alarm Time ");
      lcd.setCursor(0,1);
      lcd.print(H);
      lcd.print(":");
      lcd.print(M);
      lcd.print(":");
      lcd.print(00);
      delay(10);


      hours1_B=H;
       EEPROM.write(add_B++,hours1_B);
      minut_B=M;
      EEPROM.write(add_B++, minut_B);
      temp=0;


   delay(2000);
}
/*****************************/
void time_C(int H,int M )
{
  if(H >=24)
    {
```

```
    H=0;

    }

  if(M >=60)

    {

     M=0;

    }

    lcd.clear();

    lcd.setCursor(0,0);

    lcd.print("Set Alarm Time ");

    lcd.setCursor(0,1);

    lcd.print(H);

    lcd.print(":");

    lcd.print(M);

    lcd.print(":");

    lcd.print(00);

    delay(10);


    hours1_C=H;

     EEPROM.write(add_C++,hours1_C);

    minut_C=M;

    EEPROM.write(add_C++, minut_C);

    temp=0;


  delay(2000);

}
/****************************/

void time_D(int H,int M )
```

```
{
  if(H >=24)
    {
      H=0;
    }
  if(M >=60)
    {
      M=0;
    }
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("Set Alarm Time ");
    lcd.setCursor(0,1);
    lcd.print(H);
    lcd.print(":");
    lcd.print(M);
    lcd.print(":");
    lcd.print(00);
    delay(10);


    hours1_D=H;
     EEPROM.write(add_D++,hours1_D);
    minut_D=M;
    EEPROM.write(add_D++, minut_D);
    temp=0;
   delay(2000);
}
```

```
/************************************************/

// Function to detect alarm is occur

void match_A()

{

  int tem[17];

  for(int i=11;i<17;i++)

   {

    tem[i]=EEPROM.read(i);

   }

  if(HOUR == tem[11] && MINUT == tem[12])

  {

     alarm_A_Activated=true;

     lcd.clear();

     delay (10);

     for(int j=0;j<=17;j++)

     {

      beep();

      lcd.clear();

      lcd.setCursor(0,0);

      lcd.print("Time U'r medicine ");

      lcd.setCursor(1,1);

      lcd.print(">> # 1 ");

      beep();

      delay (500);

      digitalWrite(redLED, HIGH);

      delay(800);

      digitalWrite(redLED, LOW);
```

```
      delay (400);

     }

    digitalWrite(room_A,HIGH);

    //massage_A =("Time your medicine #1.fatima yasser 7/12/2017");

    //sendSMS(massage_A);

     delay(800);

    enterPassword();

  }

}
/***************************/

void match_B()

{

 int tem[61];

 for(int i=55;i<61;i++)

 {

   tem[i]=EEPROM.read(i);

 }

 if(HOUR == tem[55] && MINUT == tem[56])

 {

    alarm_B_Activated=true;

    lcd.clear();

    delay (10);

    for(int j=0;j<=17;j++)

    {

     beep();

     lcd.clear();

     lcd.setCursor(0,0);
```

```
        lcd.print("Time U'r medicine ");

        lcd.setCursor(1,1);

        lcd.print(">> # 2 ");

        beep();

        delay (500);

        digitalWrite(redLED, HIGH);

        delay(800);

        digitalWrite(redLED, LOW);

        delay (400);

       }

     digitalWrite(room_B,HIGH);

     //massage_B =("Time your medicine #2.walid isaed 1/12/2017");

     //sendSMS(massage_B);

      delay(800);

     enterPassword();

  }

}
/***************************/

void match_C()

{

 int tem[86];

 for(int i=80;i<86;i++)

 {

  tem[i]=EEPROM.read(i);

 }

 if(HOUR == tem[80] && MINUT == tem[81])

 {
```

```
      alarm_C_Activated=true;

    lcd.clear();

    delay (10);

    for(int j=0;j<=17;j++)

     {

      beep();

      lcd.clear();

      lcd.setCursor(0,0);

      lcd.print("Time U'r medicine ");

      lcd.setCursor(1,1);

      lcd.print(">> # 3 ");

      beep();

      delay (500);

      digitalWrite(redLED, HIGH);

      delay(800);

      digitalWrite(redLED, LOW);

      delay (400);

     }

    digitalWrite(room_C,HIGH);

    //massage_C =("Time your medicine #3.walid isaed 29/11/2017");

    //sendSMS(massage_C);

     delay(800);

    enterPassword();

 }

}

/*****************************/

void match_D()
```

```
{
 int tem[106];

 for(int i=100;i<106;i++)

  {

   tem[i]=EEPROM.read(i);

  }

 if(HOUR == tem[100] && MINUT == tem[101])

  {

     alarm_D_Activated=true;

     lcd.clear();

     delay (10);

     for(int j=0;j<=17;j++)

      {

       beep();

       lcd.clear();

       lcd.setCursor(0,0);

       lcd.print("Time U'r medicine ");

       lcd.setCursor(1,1);

       lcd.print(">> # 4 ");

       beep();

       delay (500);

       digitalWrite(redLED, HIGH);

       delay(800);

       digitalWrite(redLED, LOW);

       delay (400);

      }

     digitalWrite(room_D,HIGH);
```

```
    //massage_D =("Time your medicine #4.walid isaed 29/11/2017");

    //sendSMS(massage_D);

     delay(800);

    enterPassword();

  }

}
/************************************************/
//These function use to set alarm time (houre ,Minute)
// by using Kypade Buttons
void Time_medicine ()
{
 tone(buzzer, 2000, 100);
 bool Hum =true;
 int Temp =1;


  while (Hum)
   {
     char set_mad = myKeypad.getKey();
     if( set_mad =='A')
       {
        for (int i = 11 ; i < 17 ; i++)
          {
            EEPROM.write(i, 0);
          }
         if( set_mad =='A')
       {
          bool time_HA_activated =false;
```

```
bool time_MA_activated =false;

tone(buzzer, 2000, 100);

bool tim = true;

while (tim)

{

   //String time_HA ,time_MA;

   int k=4;

   time_HA ="";

   time_MA ="";

   time_HA_activated = true;

   while (time_HA_activated)

   {

      lcd.clear();

      lcd.setCursor(0,0);

      lcd.print("ALARM Medicine#1");

      lcd.setCursor(0,1);

      lcd.print("H >>");


       keypressed = myKeypad.getKey();

       if (keypressed != NO_KEY)

        {

            if (keypressed == '0' || keypressed == '1' || keypressed == '2' ||
keypressed == '3' ||

               keypressed == '4' || keypressed == '5' || keypressed == '6' ||
keypressed == '7' ||

               keypressed == '8' || keypressed == '9' )

            {

              tone(buzzer, 2000, 100);
```

```
            time_HA += keypressed;

            lcd.setCursor(k,1);

            lcd.print(time_HA);

            k+1;

            HA =time_HA.toInt();

        }


    }
  if ( keypressed == '*')
    {


        lcd.clear();

        lcd.setCursor(0,0);

        lcd.print("ALARM Medicine#1");

        lcd.setCursor(0,1);

        lcd.print("H >>");

        lcd.setCursor(4,1);

        lcd.print(HA);

        delay(1000);

        time_HA_activated=false;

    }
  if (keypressed =='#')
    {
      time_HA_activated=false;

      //tim =false;

    }
    delay (60);
```

```
        }
      time_MA_activated = true;
      while(time_MA_activated)
      {
         lcd.clear();
         lcd.setCursor(0,0);
         lcd.print("ALARM Medicine#1");
         lcd.setCursor(0,1);
         lcd.print("M >>");


         keypressed = myKeypad.getKey();
         if (keypressed != NO_KEY)
          {
             if (keypressed == '0' || keypressed == '1' || keypressed == '2' ||
keypressed == '3' ||
                 keypressed == '4' || keypressed == '5' || keypressed == '6' ||
keypressed == '7' ||
                 keypressed == '8' || keypressed == '9' )
             {
               tone(buzzer, 2000, 100);
               time_MA += keypressed;
               lcd.setCursor(k,1);
               lcd.print(time_MA);
               k+1;
               MA =time_MA.toInt();
             }

          }
```

```
       if ( keypressed == '*')

         {

           lcd.clear();

           lcd.setCursor(0,0);

           lcd.print("ALARM Medicine#1");

           lcd.setCursor(0,1);

           lcd.print("M >>");

           lcd.setCursor(4,1);

           lcd.print(MA);

           delay(1000);

           time_MA_activated=false;

         }

         if (keypressed =='#')

         {

           time_MA_activated=false;

            //tim =false;

         }


        delay (60);

       }

      if (keypressed =='#')

         {

           time_A (HA,MA );

            tim =false;

         }

    }

} }
```

```
/***************************/
    if( set_mad =='B')
      {
       for (int i = 55 ; i < 61 ; i++)
        {
          EEPROM.write(i, 0);
        }
        bool time_HB_activated =false;
        bool time_MB_activated =false;
        tone(buzzer, 2000, 100);
        bool tim = true;
        while (tim)
        {
          //String time_HB ,time_MB;
          int k=4;
          time_HB ="";
          time_MB ="";
          time_HB_activated = true;
          while (time_HB_activated)
          {
            lcd.clear();
            lcd.setCursor(0,0);
            lcd.print("ALARM Medicine#2");
            lcd.setCursor(0,1);
            lcd.print("H >>");
```

```
keypressed = myKeypad.getKey();

if (keypressed != NO_KEY)

{

    if (keypressed == '0' || keypressed == '1' || keypressed == '2' ||
keypressed == '3' ||

        keypressed == '4' || keypressed == '5' || keypressed == '6' ||
keypressed == '7' ||

        keypressed == '8' || keypressed == '9' )

    {

      tone(buzzer, 2000, 100);

      time_HB += keypressed;

      lcd.setCursor(k,1);

      lcd.print(time_HB);

      k+1;

      HB =time_HB.toInt();

    }


}

if ( keypressed == '*')

{


    lcd.clear();

    lcd.setCursor(0,0);

    lcd.print("ALARM Medicine#2");

    lcd.setCursor(0,1);

    lcd.print("H >>");

    lcd.setCursor(4,1);

    lcd.print(HB);
```

```
                delay(1000);

                time_HB_activated=false;

              }

          if (keypressed =='#')

            {

              time_HB_activated=false;

              //tim =false;

            }

          delay (60);

        }

        time_MB_activated = true;

        while(time_MB_activated)

        {

            lcd.clear();

            lcd.setCursor(0,0);

            lcd.print("ALARM Medicine#2");

            lcd.setCursor(0,1);

            lcd.print("M >>");


            keypressed = myKeypad.getKey();

            if (keypressed != NO_KEY)

            {

              if (keypressed == '0' || keypressed == '1' || keypressed == '2' ||
keypressed == '3' ||

                    keypressed == '4' || keypressed == '5' || keypressed == '6' ||
keypressed == '7' ||

                    keypressed == '8' || keypressed == '9' )

                {
```

```
            tone(buzzer, 2000, 100);

            time_MB += keypressed;

            lcd.setCursor(k,1);

            lcd.print(time_MB);

            k+1;

            MB =time_MB.toInt();

          }


      }
    if ( keypressed == '*')
      {


        lcd.clear();

        lcd.setCursor(0,0);

        lcd.print("ALARM Medicine#2");

        lcd.setCursor(0,1);

        lcd.print("M >>");

        lcd.setCursor(4,1);

        lcd.print(MB);

        delay(1000);

        time_MB_activated=false;

      }
   if (keypressed =='#')
     {
       time_MB_activated=false;

      //tim =false;

     }
```

```
          delay (60);

        }

      if (keypressed =='#')

          {

           time_B (HB,MB );

            tim =false;

          }

      }

   }
/***********************************/

    if( set_mad =='C')

      {

       for (int i = 80 ; i < 86 ; i++)

        {

          EEPROM.write(i, 0);

        }

       bool time_HC_activated =false;

       bool time_MC_activated =false;

       tone(buzzer, 2000, 100);

       bool tim = true;

       while (tim)

       {

         //String time_HA ,time_MA;

         int k=4;

         time_HC ="";

         time_MC ="";

         time_HC_activated = true;
```

74

```
            while (time_HC_activated)

            {

              lcd.clear();

              lcd.setCursor(0,0);

              lcd.print("ALARM Medicine#3");

              lcd.setCursor(0,1);

              lcd.print("H >>");


              keypressed = myKeypad.getKey();

              if (keypressed != NO_KEY)

               {

                  if (keypressed == '0' || keypressed == '1' || keypressed == '2' ||
keypressed == '3' ||

                     keypressed == '4' || keypressed == '5' || keypressed == '6' ||
keypressed == '7' ||

                     keypressed == '8' || keypressed == '9' )

                 {

                  tone(buzzer, 2000, 100);

                  time_HC += keypressed;

                  lcd.setCursor(k,1);

                  lcd.print(time_HC);

                  k+1;

                  HC =time_HC.toInt();

                 }


               }

             if ( keypressed == '*')

              {
```

```
            lcd.clear();

            lcd.setCursor(0,0);

            lcd.print("ALARM Medicine#3");

            lcd.setCursor(0,1);

            lcd.print("H >>");

            lcd.setCursor(4,1);

            lcd.print(HC);

            delay(1000);

            time_HC_activated=false;

        }

    if (keypressed =='#')

        {

          time_HC_activated=false;

          //tim =false;

        }

      delay (60);

  }

time_MC_activated = true;

while(time_MC_activated)

  {

     lcd.clear();

     lcd.setCursor(0,0);

     lcd.print("ALARM Medicine#3");

     lcd.setCursor(0,1);

     lcd.print("M >>");
```

```
keypressed = myKeypad.getKey();

if (keypressed != NO_KEY)

 {

   if (keypressed == '0' || keypressed == '1' || keypressed == '2' ||
keypressed == '3' ||

             keypressed == '4' || keypressed == '5' || keypressed == '6' ||
keypressed == '7' ||

             keypressed == '8' || keypressed == '9' )

    {

      tone(buzzer, 2000, 100);

      time_MC += keypressed;

      lcd.setCursor(k,1);

      lcd.print(time_MC);

      k+1;

      MC =time_MC.toInt();

    }

  }

if ( keypressed == '*')

 {

   lcd.clear();

   lcd.setCursor(0,0);

   lcd.print("ALARM Medicine#3");

   lcd.setCursor(0,1);

   lcd.print("M >>");

   lcd.setCursor(4,1);

   lcd.print(MC);

   delay(1000);
```

```
                    time_MC_activated=false;

                }
            if (keypressed =='#')

                {

                 time_MC_activated=false;

                 //tim =false;

                }
             delay (60);

            }
          if (keypressed =='#')

            {

             time_C (HC,MC);

             tim =false;

            }

          }

      }


/*********************************/

    if( set_mad =='D')

      {

      for (int i = 90 ; i < 96 ; i++)

        {

          EEPROM.write(i, 0);

        }
        bool time_HD_activated =false;

        bool time_MD_activated =false;

        tone(buzzer, 2000, 100);
```

```
bool tim = true;

while (tim)

{

  //String time_HA ,time_MA;

  int k=4;

  time_HD ="";

  time_MD ="";

  time_HD_activated = true;

  while (time_HD_activated)

  {

    lcd.clear();

    lcd.setCursor(0,0);

    lcd.print("ALARM Medicine#4");

    lcd.setCursor(0,1);

    lcd.print("H >>");


    keypressed = myKeypad.getKey();

    if (keypressed != NO_KEY)

    {

      if (keypressed == '0' || keypressed == '1' || keypressed == '2' ||
keypressed == '3' ||

        keypressed == '4' || keypressed == '5' || keypressed == '6' ||
keypressed == '7' ||

        keypressed == '8' || keypressed == '9' )

      {

        tone(buzzer, 2000, 100);

        time_HD += keypressed;

        lcd.setCursor(k,1);
```

```
            lcd.print(time_HD);

            k+1;

            HD =time_HD.toInt();

          }


      }

    if ( keypressed == '*')

      {

        lcd.clear();

        lcd.setCursor(0,0);

        lcd.print("ALARM Medicine#4");

        lcd.setCursor(0,1);

        lcd.print("H >>");

        lcd.setCursor(4,1);

        lcd.print(HD);

        delay(1000);

        time_HD_activated=false;

      }

  if (keypressed =='#')

    {

      time_HD_activated=false;

      //tim =false;

    }

    delay (60);

  }

time_MD_activated = true;

while(time_MD_activated)
```

```
{
    lcd.clear();

    lcd.setCursor(0,0);

    lcd.print("ALARM Medicine#4");

    lcd.setCursor(0,1);

    lcd.print("M >>");


    keypressed = myKeypad.getKey();

    if (keypressed != NO_KEY)

     {

        if (keypressed == '0' || keypressed == '1' || keypressed == '2' ||
keypressed == '3' ||

            keypressed == '4' || keypressed == '5' || keypressed == '6' ||
keypressed == '7' ||

            keypressed == '8' || keypressed == '9' )

        {

            tone(buzzer, 2000, 100);

            time_MD += keypressed;

            lcd.setCursor(k,1);

            lcd.print(time_MD);

            k+1;

            MD =time_MD.toInt();

        }


     }

    if ( keypressed == '*')

     {

        lcd.clear();
```

```
                lcd.setCursor(0,0);

                lcd.print("ALARM Medicine#4");

                lcd.setCursor(0,1);

                lcd.print("M >>");

                lcd.setCursor(4,1);

                lcd.print(MD);

                delay(1000);

                time_MD_activated=false;

              }

           if (keypressed =='#')

              {

                time_MD_activated=false;

                //tim =false;

              }

            delay (60);

           }

         if (keypressed =='#')

            {

              time_D (HD,MD );

              tim =false;

            }

        }

     }


 if (set_mad == '#')

   {

    tone(buzzer, 2000, 100);
```

```
        Temp =0;

        Hum =false;

      }



    }

}




/*************************************************/

// This function for Temperature _humidity

//Read sensore value and display on LCD

void Temp_humidity ()

{

  int chk = DHT.read11(DHT11_PIN);

  lcd.clear();

  lcd.setCursor(0,0);

  lcd.print("Temp: ");

  lcd.print(DHT.temperature);

  lcd.print((char)223);

  lcd.print("C");

  lcd.setCursor(0,1);

  lcd.print("Humidity: ");

  lcd.print(DHT.humidity);

  lcd.print("%");

  delay(1400);

/******************/

//
```

```
if( DHT.temperature >= 27)

 {

  digitalWrite (fan,HIGH);

 }

 if (DHT.temperature < 27)

 {

  digitalWrite (fan,LOW);

 }

}
/*************************************************/
//This function Display : Name & Date medicine

void medicine ()

{

 tone(buzzer, 2000, 100);

 int med =1;

 if (med)

 {

  bool medicie =true;

  while (medicie)

  {

   whichKey = myKeypad.getKey();

   if (whichKey == 'A')

    {

     tone(buzzer, 2000, 100);

     lcd.clear();

     lcd.setCursor(0,0);

     lcd.print("Name:Ahmad  ");
```

```
   lcd.setCursor(0,1);

   lcd.print("Date: 12 ");

   delay (2000);

  }

if (whichKey == 'B')

  {

  tone(buzzer, 2000, 100);

  lcd.clear();

  lcd.setCursor(0,0);

  lcd.print("Name: mohammad  ");

  lcd.setCursor(0,1);

  lcd.print("Date: 13 ");

  delay (2000);

  }

if (whichKey == 'C')

  {

  tone(buzzer, 2000, 100);

  lcd.clear();

  lcd.setCursor(0,0);

  lcd.print("Name: omar  ");

  lcd.setCursor(0,1);

  lcd.print("Date: 14  ");

  delay (2000);

  }

if (whichKey == 'D')

  {

  tone(buzzer, 2000, 100);
```

```
        lcd.clear();

        lcd.setCursor(0,0);

        lcd.print("Name: jak ");

        lcd.setCursor(0,1);

        lcd.print("Date: 15 ");

        delay (2000);

        }

      if (whichKey == '#')

       {

        tone(buzzer, 2000, 100);

        med =0;

        medicie =false;

       }



    }



 }

}
/*************************************************/

//This function Display : Time of medicine alarm

void Display()

{

 tone(buzzer, 2000, 100);

 int Dis =1;

 if (Dis)

 {

   bool Display =true;
```

```
while (Display)

 {

  whichKey = myKeypad.getKey();

 if (whichKey == 'A')

   {

     tone(buzzer, 2000, 100);

     lcd.clear();

     lcd.setCursor(0,0);

     lcd.print("Time Medicine #1");

     lcd.setCursor(0,1);

     lcd.print("Date:");

     lcd.setCursor(6,1);

     lcd.print(HA);

     lcd.setCursor(8,1);

     lcd.print(":");

     lcd.print(MA);

     delay (2000);

   }

 if (whichKey == 'B')

   {

     tone(buzzer, 2000, 100);

     lcd.clear();

     lcd.setCursor(0,0);

     lcd.print("Time Medicine #2");

     lcd.setCursor(0,1);

     lcd.print("Date:");

     lcd.setCursor(6,1);
```

```
      lcd.print(HB);

      lcd.setCursor(8,1);

      lcd.print(":");

      lcd.print(MB);

      delay (2000);

    }

  if (whichKey == 'C')

    {

      tone(buzzer, 2000, 100);

      lcd.clear();

      lcd.setCursor(0,0);

      lcd.print("Time Medicine #3");

      lcd.setCursor(0,1);

      lcd.print("Date:");

      lcd.setCursor(6,1);

      lcd.print(HC);

      lcd.setCursor(8,1);

      lcd.print(":");

      lcd.print(MC);

      delay (2000);

    }

  if (whichKey == 'D')

    {

      tone(buzzer, 2000, 100);

      int COUNT_H = EEPROM.read(11);

      int COUNT_M = EEPROM.read(12);

      int COUNT_B = EEPROM.read(55);
```

```
int COUNT_S = EEPROM.read(56);


int COUNT_q = EEPROM.read(80);

int COUNT_w = EEPROM.read(81);

int COUNT_e = EEPROM.read(100);

int COUNT_r = EEPROM.read(101);

lcd.clear();

lcd.setCursor(0,0);

lcd.print(COUNT_H);

lcd.setCursor(2,0);

lcd.print(":");

lcd.setCursor(3,0);

lcd.print(COUNT_M);

lcd.setCursor(6,0);

lcd.print("|");

lcd.setCursor(7,0);

lcd.print(COUNT_B);

lcd.setCursor(9,0);

lcd.print(":");

lcd.setCursor(10,0);

lcd.print(COUNT_S);


lcd.setCursor(0,1);

lcd.print(COUNT_q);

lcd.setCursor(2,1);

lcd.print(":");

lcd.setCursor(3,1);
```

```
        lcd.print(COUNT_w);

        lcd.setCursor(6,1);

        lcd.print("|");

        lcd.setCursor(7,1);

        lcd.print(COUNT_e);

        lcd.setCursor(9,1);

        lcd.print(":");

        lcd.setCursor(10,1);

        lcd.print(COUNT_r);

     }

 if (whichKey == '*')

 {

        tone(buzzer, 2000, 100);

        lcd.clear();

        lcd.setCursor(0,0);

        lcd.print(Count_A);

        lcd.setCursor(3,1);

        lcd.print(Count_B);

        lcd.setCursor(6,0);

        lcd.print(Count_C);

 }


   if (whichKey == '#')

    {

      tone(buzzer, 2000, 100);

      Dis =0;

      Display =false;
```

```
      }


    }


  }
  }
/**********************************************/

void sendSMS(String massage)

{

 // AT command to set SIM900 to SMS mode

 SIM900.print("AT+CMGF=1\r");

 delay(100);


 // REPLACE THE X's WITH THE RECIPIENT'S MOBILE NUMBER

 // USE INTERNATIONAL FORMAT CODE FOR MOBILE NUMBERS

 SIM900.println("AT + CMGS = \"+972597362791\"");

 delay(100);


 // REPLACE WITH YOUR OWN SMS MESSAGE CONTENT

 SIM900.println(massage);

 delay(100);


 // End AT command with a ^Z, ASCII code 26

 SIM900.println((char)26);

 delay(100);

 SIM900.println();

 // Give module time to send SMS
```

```
  delay(5000);

}

/***********************************************/

//open correct room when alarme is occur and decrement

//medicine counter by one "1"


void your_time()

{

/********Room # 1***********/

 if (alarm_A_Activated)

  {

   Password_Acti = false;

   lcd.clear();

   lcd.setCursor(0, 0);

   lcd.print(" Go to Room # 1");

   delay (2000);

   beep();

   /******************/

   // open room # 1

     servo_A.write(50);

    delay (5000);

   // Close room # 1

     servo_A.write(90);

     delay(15);


   /******************/

   delay(500);
```

```
    lcd.clear();

    Count_A--;

    digitalWrite(room_A,LOW);

    delay(500);

    alarm_A_Activated=false;

  }
/********Room # 2***********/

 if (alarm_B_Activated)

  {

    Password_Acti = false;

    lcd.clear();

    lcd.setCursor(0, 0);

    lcd.print(" Go to Room # 2");

    delay (2000);

    beep();

    /******************/

    // open room # 2

    for (pos = 0; pos <= 90; pos += 5)

     {

       servo_A.write(pos);

       delay(15);

      }

     delay (5000);

     // Close room # 2

    for (pos = 90; pos >= 0; pos -= 5)

     {

       servo_A.write(pos);
```

```
    delay(15);
   }
/*****************/
  delay(500);
  lcd.clear();
  Count_B--;
  //counter ( Count_A,Count_B,Count_C);
  digitalWrite(room_B,LOW);
  delay(500);
  alarm_B_Activated=false;
 }
/********Room # 3***********/
 if (alarm_C_Activated)
  {
   Password_Acti = false;
   lcd.clear();
   lcd.setCursor(0, 0);
   lcd.print(" Go to Room # 3");
   delay (2000);
   beep();
   /*****************/
   // open room # 3


     servo_A.write(85);
    delay (5000);
    // Close room # 3
     servo_A.write(180);
```

```
        /*****************/

        delay(500);

        lcd.clear();

        Count_C--;

        digitalWrite(room_C,LOW);

        delay(500);

        alarm_C_Activated=false;

    }

}


/************************************************/

// Eenter Password to Open room medicine

void enterPassword()

{

  tone(buzzer, 2000, 100);

  int k=5;

  tempPassword = "";

  Password_Acti = true;

  lcd.clear();

  lcd.setCursor(0,0);

  lcd.print(" Enter Password ");

  lcd.setCursor(0,1);

  lcd.print("Pass>");

   while(Password_Acti)

    {

       keypressed = myKeypad.getKey();
```

```
    if (keypressed != NO_KEY)

    {

      if (keypressed == '0' || keypressed == '1' || keypressed == '2' || keypressed == '3'
||

        keypressed == '4' || keypressed == '5' || keypressed == '6' || keypressed == '7' ||

        keypressed == '8' || keypressed == '9' )

      {

        tone(buzzer, 2000, 100);

        tempPassword += keypressed;

        lcd.setCursor(k,1);

        lcd.print("*");

        k++;

      }

    }

    if (k > 9 || keypressed == '#')

    {

      tempPassword = "";

      k=5;

      lcd.clear();

      lcd.setCursor(0,0);

      lcd.print(" Enter Password ");

      lcd.setCursor(0,1);

      lcd.print("Pass>");

    }

    if ( keypressed == '*')

    {

      if ( tempPassword == password )
```

```
{

   tone(buzzer, 2000, 100);

   lcd.clear();


     lcd.clear();

     lcd.setCursor(0, 0);

     lcd.print("Correct Password");

     lcd.setCursor(0, 1);

     lcd.print("Get U'r Medicin");

     delay (2000);

   your_time();

}

if (tempPassword == medPassword )

{

 tone(buzzer, 2000, 100);

 lcd.clear();

 lcd.clear();

 lcd.setCursor(0, 0);

 lcd.print("Correct Password");

 delay (1200);

 lcd.setCursor(0, 0);

 lcd.print("This to filling ");

 lcd.setCursor(0, 1);

 lcd.print(" Medicin in Box");

 delay (3000);

 /*************/

  servo_A.write(85);// open room # 1
```

```
        servo_B.write(85);// open room # 2

        servo_C.write(85);// open room # 3

        servo_D.write(85);// open room # 4

        delay (800);

   if ( keypressed == '*')

    {

      servo_A.write(180);// CLOSE room # 1

      servo_B.write(180);// CLOSE room # 2

      servo_C.write(180);// CLOSE room # 3

      servo_D.write(180);// CLOSE room # 4

      delay (800);

    }

   else if (tempPassword != password)

   {

    lcd.setCursor(0,0);

    lcd.print("Wrong! Try Again");

    delay(2000);

    lcd.clear();

    lcd.setCursor(0,0);

    lcd.print(" Enter Password ");

    lcd.setCursor(0,1);

    lcd.print("Pass>");

   }

  }

 }

}
```

```
/*************************************************/

// Counter function and RGB LED lighting

//Counter for room #1 *******************
void counter_A (int Count_A)
  {
    if (Count_A >7 /*&& Count_A <=10*/ )
     {
      digitalWrite (G_A,HIGH);
      digitalWrite (R_A,LOW);
      digitalWrite (B_A,LOW);
     }
    if (Count_A >=4 && Count_A <=7 )
     {
      digitalWrite (B_A,HIGH);
      digitalWrite (G_A,LOW);
      digitalWrite (R_A,LOW);
     }
    if (Count_A >=0 && Count_A <=3 )
     {
      digitalWrite (R_A,HIGH);
      digitalWrite (G_A,LOW);
      digitalWrite (B_A,LOW);
     }
  }
//Counter for room #2 *******************
void counter_B (int Count_B)
  {
```

99

```
if (Count_B >7 && Count_B <=10 )

 {

 digitalWrite (G_B,HIGH);

 digitalWrite (R_B,LOW);

 digitalWrite (B_B,LOW);

 }

if (Count_B >=4 && Count_B <=7 )

 {

 digitalWrite (B_B,HIGH);

 digitalWrite (G_B,LOW);

 digitalWrite (R_B,LOW);

 }

if (Count_B >=0 && Count_B <=3 )

 {

 digitalWrite (R_B,HIGH);

 digitalWrite (G_B,LOW);

 digitalWrite (B_B,LOW);

 }

 }

//Counter for room #3 *******************

void counter_C (int Count_C)

 {

 if (Count_C >7 && Count_C <=10 )

 {

 digitalWrite (G_C,HIGH);

 digitalWrite (R_C,LOW);

 digitalWrite (B_C,LOW);
```

```
      }
    if (Count_C >=4 && Count_C <=7 )

    {

     digitalWrite (B_C,HIGH);

     digitalWrite (G_C,LOW);

     digitalWrite (R_C,LOW);

     }
    if (Count_C >=0 && Count_C <=3 )

    {

     digitalWrite (R_C,HIGH);

     digitalWrite (G_C,LOW);

     digitalWrite (B_C,LOW);

     }

}
//Counter for room #4 ******************

void counter_D (int Count_D)

 {

    if (Count_D >7 && Count_D <=10 )

    {

     digitalWrite (G_D,HIGH);

     digitalWrite (R_D,LOW);

     digitalWrite (B_D,LOW);

     }

    if (Count_D >=4 && Count_D <=7 )

    {

     digitalWrite (B_D,HIGH);

     digitalWrite (G_D,LOW);
```

```
    digitalWrite (R_D,LOW);

    }

    if (Count_D >=0 && Count_D <=3 )

    {

    digitalWrite (R_D,HIGH);

    digitalWrite (G_D,LOW);

    digitalWrite (B_D,LOW);

    }

}


/*************************************************/
```

Chapter 7

Results and Conclusions

Results that obtained due to system implementation and testing are observed and recorded in this chapter. These results are recorded according to test procedures and give a good impact about the system behavior . The results are studied and analyzed providing an important conclusions mentioned in this chapter.

7.1 Sensor Unit Results

DHT 11 Sensor response is very sensitive to humidity and temperature . When the humidity or temperature exceeds the limit required for keeping the medicine inside the pharmacy, the sensor senses it and sends a signal to the controller until it is processed and then send a signal to the fan until you turn to save the medicine.

7.2 Prototype Operation Results

All system components are combined to get whole prototype and then activated. The system operates as expected and give a good results that can be summarized as follow:

1- When the person open the pharmacy , at first appears on the screen time and date.

2- if the person press(F1) , you can go to gate (F1) it have expire date of medicine , but if the person press(F2) , you can go to gate (F2) it have name of medicine an the time to take it, although if the person press(F3) , you can go to gate (F3) it have information on temperature and humidity ,also - if the person press(F4) , you can go to gate (F4) it have alarm system , and all system when press # go out from the gate and return to time and date .

7.3 Conclusions

The project helps to alert the medication on time or as the doctor prescription says, the project also prevents and saves the children from taking the medication accidently by the password through which the pharmacy is opened, the pharmacy will be animated so it will be available whenever the patient needs it, the project also reduces the costs of throwing a drug that has been damaged by an unsuitable storage situation, by monitoring the temperature inside the pharmacy and keeping it in shaded places, also it is working to alert when the expiration of the drug date occurs and warning in case of shortage of pills, in addition of Containing information and details of Drug With a message receiving to the patient mobile remembering the medicine date .mobile phone showing : medicine name ,doctor prescription , residual quantity and expiration date .

References

1. *Walkerville, Ontario, Canada, New York [etc.] H. Walker & sons, limited [1911] LC: HD9395.U45 R7*

2. *1908 Wiley, Harvey Washington. Washington, Govt. print. off., 1908. LC: S585*
   .A1 no. 84 pt. 5 Historic meeting to commemorate fortieth anniversary of original
   federal Food and drugs act.

3. *Descorides "medical material*

4. *Global report on diabetes*

5. *Traditional Medicine. "World Health Organization web*

6. *-Site.http: //www.who.int/mediacentre/factsheets/fs134/en/index.html. Accessed on 3/12/09*

7. *David Winston. American Extra Pharmacopoeia*

8. *Chinese Herbal Medicine: Materia Medica, Third Edition by Dan Bensky, Steven Clavey, Erich Stoger, and Andrew Gamble.*

9. *O'Connor, J.; Lee, E.; Rubinstein, F.; Selkowitz, S. (1997). " novel silicon nanotips antireflection surface for the micro sun sensor" . Lawrence Berkeley National Laboratory, LBNL-39945. pp. Chapter 8. Retrieved 3 September 2009.*

10. *K. Ohishi, K. Ohnishi, K. Miyachi, "Microprocessor-used speed control of dc motor without speed sensor", Conf. Rec. IECON' 84, pp. 468-473, 1984.*

11. *Rubinstein, F.; Ward, G.; Verderber, R. (1989). "Intelligent cooling fan" (PDF). Journal of the Illuminating Engineering Society, 18(1). pp. 70–94. Retrieved 6 September 2009.*

12. *National Grid (2009). "Memory control system" . National Grid. Retrieved 3 September 2009.*

13. *Birt, B.; Newsham, G. R. (2009). "Thin keypad assembly").   11 June 2011. Retrieved 3 September 2009.*

14. *M. Spong, "The swing up control problem for the acrobot", IEEE Control Systems Magazine, Volume 15, Number 5, pp. 49-55, 1995.*

15. *Yoo, Seunghwan; Kim, Jonghun; Jang, Cheol-Yong; Jeong, Hakgeun. A System and method for an electronic postage scale with variable function keys and window screens . Retrieved 1 September 2014.*

16. *R. B. Staszewski et al., "A First Multigigahertz Digitally Controlled Oscillator for Wireless Applications," IEEE Trans. Microwave Theory and Techniques, vol. 51, no. 11, pp. 2154-2164, Nov., 2003.*

17. *D'Ausilio, A. (2012). Arduino: A low-cost multipurpose lab equipment.Retrieved 1 September 2014.*

18. *Canto, R., Bufalari, I., & D'Ausilio, A. (2011). A convenient and accurate parallel Input/Output USB.Retrieved 15 August 2009.*

19. *EAGLE files: arduino-mega2560-reference-design.*
20. *EAGLE files: arduino-mega2560-reference-design.zip Schematic: arduino-mega2560-schematic.pdf*

21. *Davidson KW, Kahn A, Price RD: Reduction of Adverse Drug Reactions by Computerized Drug Interaction Screening. The Journal of Family Practice 1987; 25(4):371-375*

22. *Hammarlund ER, Ostrom JR, Kethley AJ: The Effects of Drug Counseling and Other Educational Strategies on Drug Utilization of the Elderly. Medical Care 1985; 23(2): 165*

23. *Britton ML, Lurvey PL: Impact of Medication Profile Review on Prescribing in a General Medicine Clinic. American Journal ofHospital Pharmacy 1991;48: 265*

24. *Gosney M, Tallis R: Prescription of Contraindicat-ed and Interacting Drugs in Elderly Patients Admittedto Hospital. Lancet 1984; 2(8402):564*