PROTOTYPE FOR IOT PET FEEDER

A MINI PROJECT REPORT

Submitted by

S.Sridharshini	1NH18EE741
Varun sham kumar	1NH18EE753
Srinjana chowduri	1NH18EE750

In partial fulfilment for the ward of the degree of

BACHELOR OF ENGINEERING

IN

ELECTRICAL AND ELECTRONICS ENGINEERING

CHAPTER 1

INTRODUCTION

Nowadays most of us are fascinated to have pets at their home. But these pets have to be taken care properly. Their feeding on time is an important task as they become part of our family. But in our busy schedule we fail to pay attention on our pet thus it doesn't get proper food on time. This paper addresses the above issue by introducing an Automatic Pet Feeding System to ensure feeding pet on frequent interval of time. Automatic Pet Feeding System consists food storage, servo motor, dispenser, feeding bowl, etc. It also features Arduino to automatically control the operations. It is also possible to make more hitech by adding cameras and audio box to check activities of pet and talk to it.

Automatic pet feeding system features a machine which can feed pets (e.g. dogs) automatically after frequent intervals in absence of his master. By using machine master don't have to stay with his pet every time to feed it and he gets liberty to do his other works outside without caring about his pet. Dish for feeding pet could be filled in number of ways one is we can set the time and date using Arduino UNO which is displayed on LCD screen fitted on body of pet feeder.

This is our automatic pet feeder powered by arduino, using a auger, and programmable with two feeding times with user set quantity of food, with a battery backed up internal clock. There are some pet feeders available in UK. They are either the type that rotate and unveil a meal, up to four times only, or the half decent machine with a hopper is hard to program, forgets everything if you remove the batteries, and frequently fails to even deliver any food. America has some good machines but they start at $300 USD, and shipping would be painful to the India, so we decided to make one.

CHAPTER 2

2.1 SCOPE OF THE PROJECT

Scope is that the range of view, perception, grasp or abilities; capacity for action in any activity, topic, project or work. it's actually the chance for activity and improves the work that anyone is doing. the most aim of this project is to scale back the human effort for fetching the medication as prescribed by the medical consultants. The further scope of this project is as follows:

I. It will be accustomed reduce the human pressure in medical stores II. It will be accustomed void errors in giving out the medication.

III. It will be accustomed separate the expired medicine within the store.

2.2 OBJECTIVE

Objective might be a selected result that a private or system aims to comprehend among a timeframe and with out there resources. There is also one or more objectives for any particular work. Objectives are basic tools that underline all the design and strategic activities. They function the thought for creating policy and evaluating performance. The objectives for our project are as follows: I. to understand the working of Arduino board.

II. to seek out out the uses of Arduino board.

III. to understand the employment of Arduino motor shield.

IV. to scale back the human effort for fetching the medication as prescribed by the medical consultants V. to scale back the human pressure in medical stores.

CHAPTER 3

3.1 TOOLS REQUIRED

1. audrino uno(the microprocessor to which the code is uploaded)
2. 4X4 matrix keypad
3. 12X2 lcd display(to display real time and check input time)
4. push button (gives audrino csignal for setting time)
5. servo button(rotates the bottom of the container to open it)
6. connecting wires(used for internal connections)
7. Bread board(all connections are made on this)

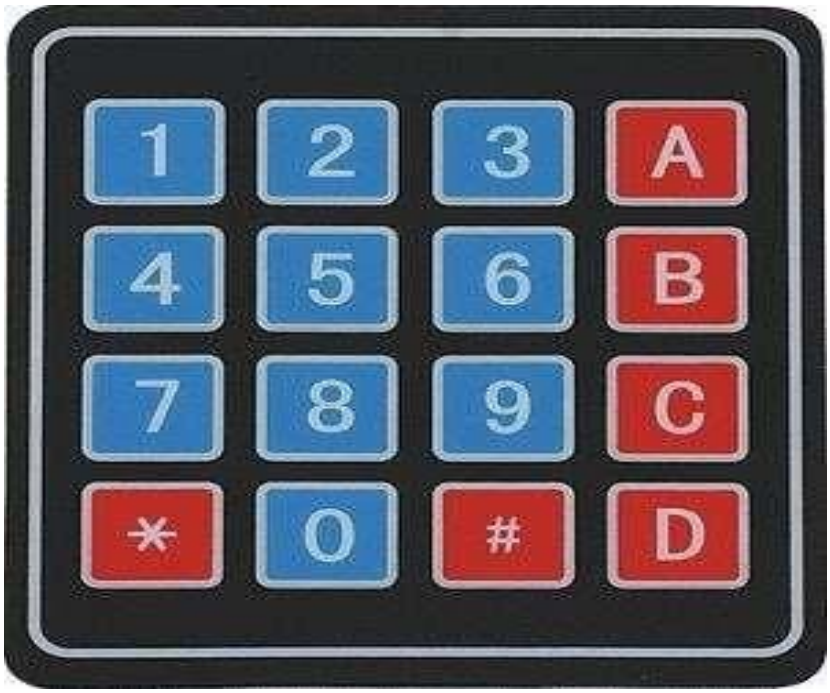
3.2 Arduino UNO

Fig (2): Arduino UNO

The Arduino UNO is a microcontroller board supporting the ATmega328. It has twenty digital input/output pins (of that half-dozen may be used as PWM outputs and half-dozen may be used as analog inputs), a sixteen rate resonator, a USB connection, an influence jack, AN incircuit system programming (ICSP) header, and a push. It contains all things required to support the microcontroller; merely connect it to a laptop and other deviceswith a USB cable or power it with a AC-to-DC adapter or battery to urge started. The ATmega328 provides UART TTL (5V) serial communication, that is obtainable on digital pins zero (RX) and one (TX). The hardware reference style is distributed below an imaginative Commons Attribution Share-Alike a pair of.5 license and is obtainable on the Arduino web site. Programs may be loaded on to that from the easy-to-use Arduino bug. The Arduino has an intensive support community, that makes it a awfully simple thanks to start operating with embedded natural philosophy. The R3 is that the third, and latest, revision of the Arduino Uno.

3.5 Keypad

This type of keypad is basic types of the keypad, which has the connected in matrix form. When a switch is entered it is connected and the number written on the key is known by using matrix.



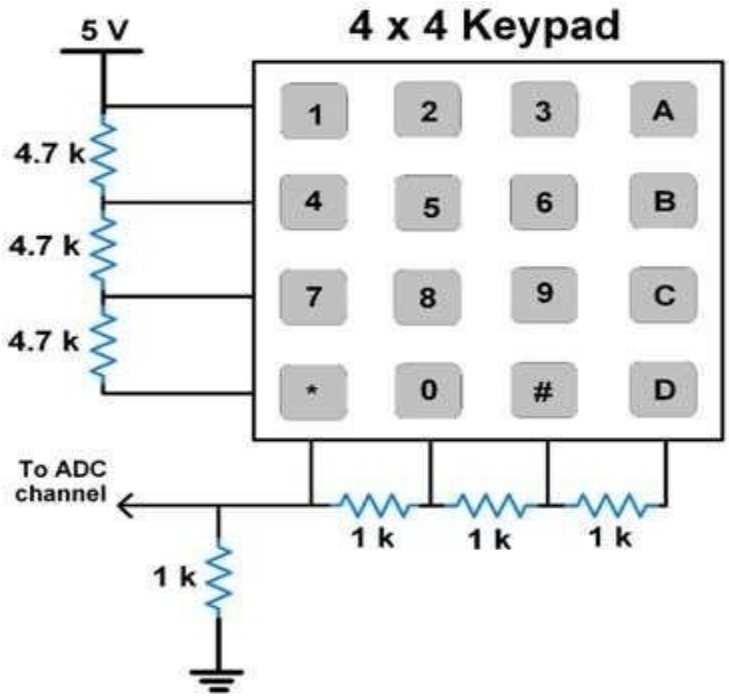
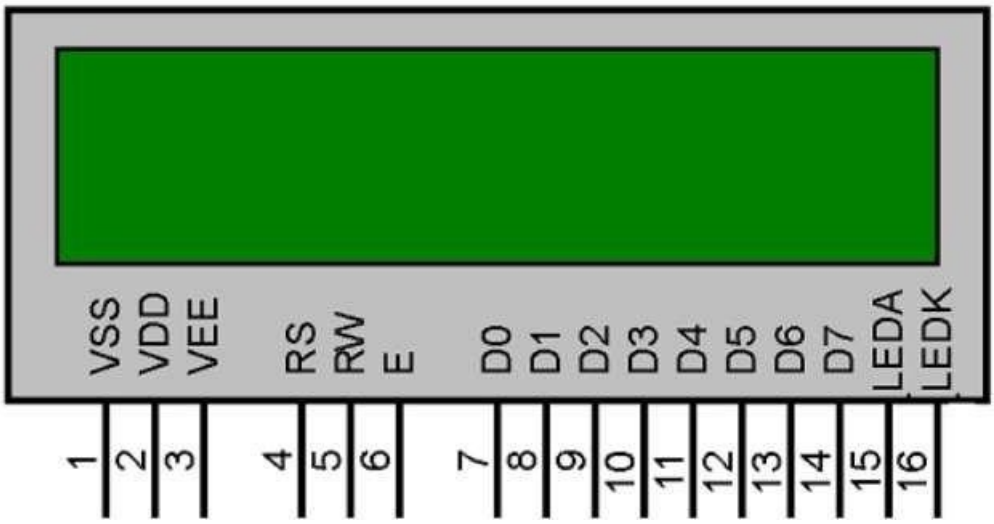


Fig (7): keypad circuit

16X2 LCD DISPLAY:



16×2 LCD is named so because; it has 16 Columns and 2 Rows. There are a lot of combinations available like, 8×1, 8×2, 10×2, 16×1, etc. But the most used one is the 16*2 LCD, hence we are using it here.

Sr. No	Pin No.	Pin Name	Pin Type	Pin Description	Pin Connection
1	Pin 1	Ground	Source Pin	This is a ground pin of LCD	Connected to the ground of the MCU/ Power source
2	Pin 2	VCC	Source Pin	This is the supply voltage pin of LCD	Connected to the supply pin of Power source
3	Pin 3	V0/VEE	Control Pin	Adjusts the contrast of the LCD.	Connected to a variable POT that can source 0-5V

4	Pin 4	Register Select	Control Pin	Toggles between Command/Data Register	Connected to a MCU pin and gets either 0 or 1. 0 -> Command Mode 1> Data Mode
5	Pin 5	Read/Write	Control Pin	Toggles the LCD between Read/Write Operation	Connected to a MCU pin and gets either 0 or 1. 0 -> Write Operation 1-> Read Operation
6	Pin 6	Enable	Control Pin	Must be held high to perform Read/Write Operation	Connected to MCU and always held high.
7	Pin 714	Data Bits (0-7)	Data/Command Pin	Pins used to send Command or data to the LCD.	In 4-Wire Mode Only 4 pins (0-3) is connected to MCU In 8-Wire Mode All 8 pins(07) are connected to MCU
8	Pin 15	LED Positive	LED Pin	Normal LED like operation to illuminate the LCD	Connected to +5V
9	Pin 16	LED Negative	LED Pin	Normal LED like operation to illuminate the LCD connected with GND.	Connected to ground

PUSH BUTTON:

Push-Buttons are normally-open **tactile switches**. Push buttons allow us to power the circuit or make any particular connection only when we press the button. Simply, it makes the circuit connected when pressed and breaks when released. A push button is also used for triggering of the SCR by gate terminal. These are the most common buttons which we see in our daily life electronic equipment's. Some of the applications of the Push button are mentioned at the end of the article.

SERVO BUTTON:

CHAPTER 4

PROJECT DESCRIPTION

4.1 Block Diagram

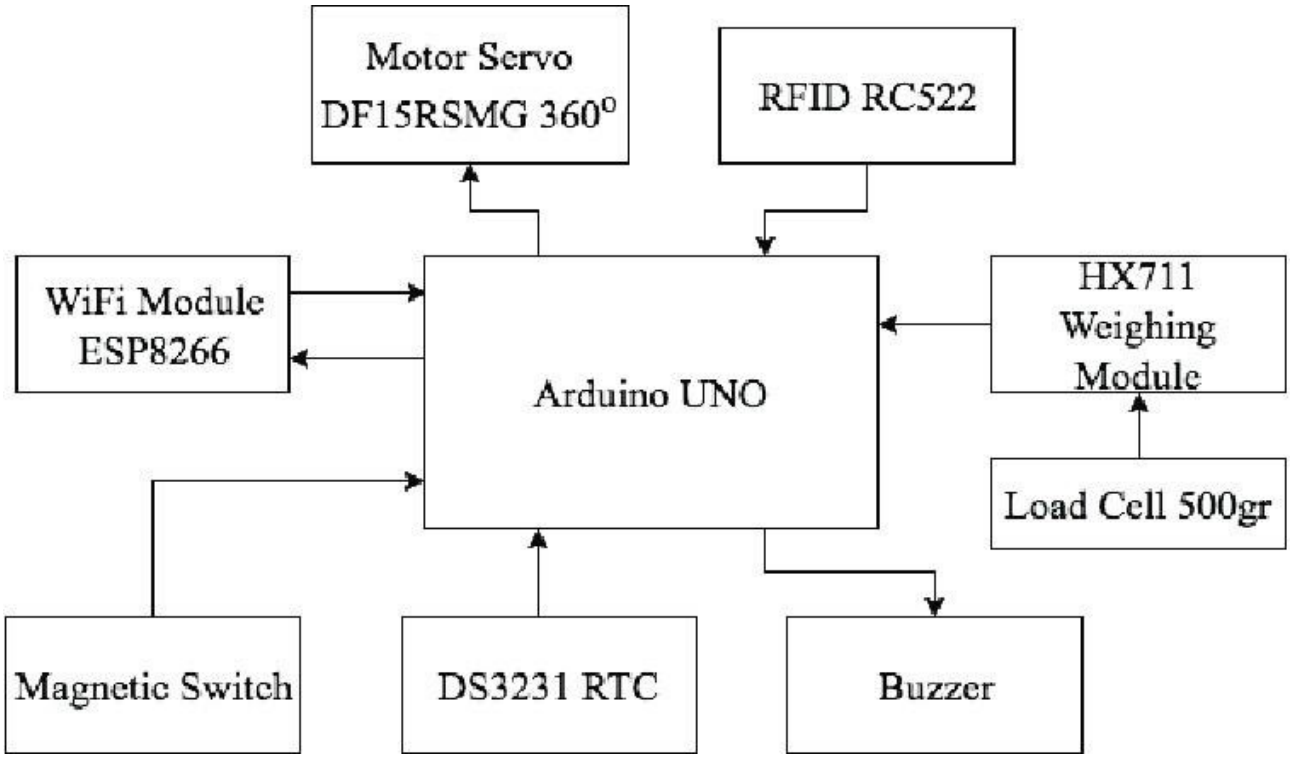


Fig (8): block diagram

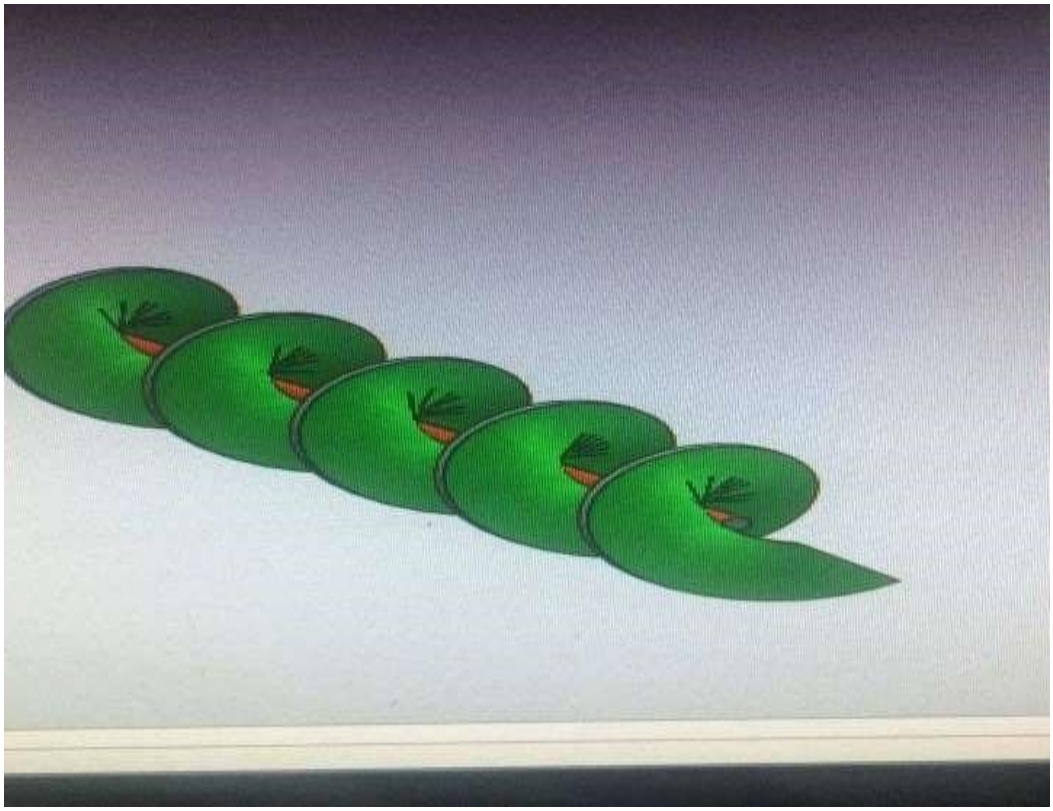
4.2 CONSTRUCTION AND WORKING

4.2.1 CONSTRUCTION

Working of the Automatic Pet Feeder is given below according to the components used in the system:

1. **Servo motor**

   We are using a servo motor which rotates 100 degrees to open the container and 55 to close it. Modulation & having the torque (for 6.0V: 152.76 oz-in), Speed (for 6.0V: 0.16 sec/60 degree), weight(1.94 oz=55.0g), Gear type(Metal). The motor rotates the AUGER which delivers food from storage to dish. Motor gets the signals from Arduino UNO which is already programmed. When AUGER starts rotating due to servo motor the food from the upper body of the mechanism transmits to the dish. The shape of auger is like a drilling device, Auger which we are using is made by 3D Printing



2. ARDUINO UNO:

Arduino UNO is a micro controller board based on the ATmega328. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP

header, and a reset button. It contains everything needed to control the microcontroller. Simply connecting with computer using USB cable to feed the program and get started. Arduino UNO is programmed by using Arduino IDE (an open source arduino software). Programming is done in a way that we can set the time and date for the system which is displayed on the LED screen. Real Time Clock (RTC) which is connected to arduino.the time is set by using Ky-040 rotary decoder encoder. When real time matches with the RTC it gives the signal to motor and motor starts rotating. Further motor gives the motion to the auger which transfers the food to the dish. Some features of Arduino UNO are as follows:

Microcontrollers:

ATmega328. Operating

Voltage: 5V.

Input Voltage

(recommended): 7-12V.

Input Voltage (limits): 6-20V.

Digital I/O Pins: 14(of which 6 provide PWM

output). Analog Input Pins: 6.

DC Current per I/O Pin: 40mA.

DC Current for

3.3V pm: 50mA.

Flash memory: 32 KB of which 0.5 KB used boot

loader. SRAM: 2KB (ATmega328).

EE PROM: 1KB

(ATmega328). Clock

speed: 16 MHz.

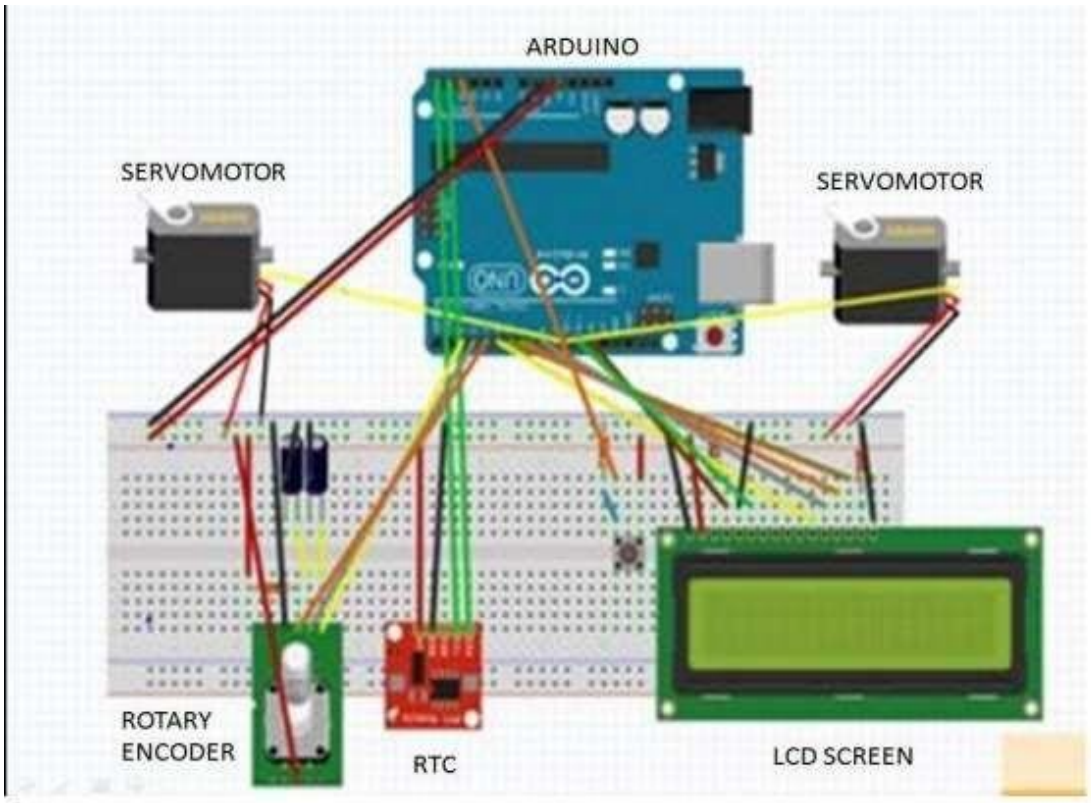


Fig. 2: Circuit Diagram of Arduino

3. 4x4 Keypad: In Automatic pet feeder we are using a 4x4 keypad which takes in information in rows and coloumn format.

When this button is pressed...	If...	Then...
1	Col0=high	Row0=high
2	Col1=high	Row0=high
3	Col2=high	Row0=high
4	Col0=high	Row1=high
5	Col1=high	Row1=high
6	Col2=high	Row1=high
7	Col0=high	Row2=high
8	Col1=high	Row2=high
9	Col2=high	Row2=high
0	Col1=high	Row3=high
*	Col0=high	Row3=high
#	Col2=high	Row3=high
A	Col3=high	Row0=high
B	Col3=high	Row1=high
C	Col3=high	Row2=high
D	Col3=high	Row3=high

MECHANISM

We are using an acrylic fibre for making the outer body of pet feeder. Acrylic fibre is a synthetic fibre of polyacrylonitrile. Acrylate polymer is a group of polymers (plastics) noted for transparency and elasticity. Acrylic is safe for making food storage containers. Acrylic is nontoxic material for making the containers for storage of food. In this, we are using a 120 cm x 120 cm acrylic sheet. The thickness of sheet is 3 mm thick. The upper body of the pet feeder i.e. container is made by 25 cm x 25 cm x 25 cm box. And the lower body of the pet feeder is made by 35 cm x 35 cm x 15 cm box. The upper body consists the container for food storage having 1 kg capacity. The lower body consists the space for electronic operating components and display for the set time and set quantity. Lower body also consists the slot for dispensing the food from the container to the bowl according to the set time and in set quantity.

METHADOLOGY

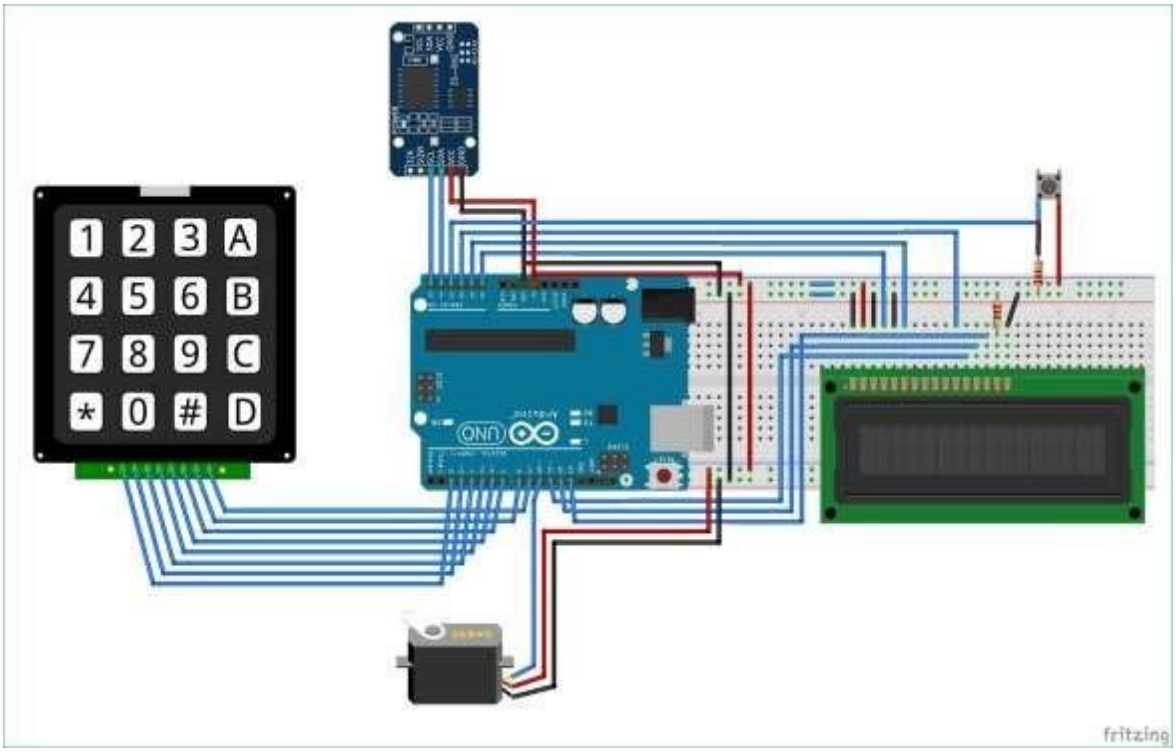
The code for light sensing is written on the arduino software. In the code the wifi specifications like ssid and password has been mentioned. At the time of updating the code from the arduino to the NodeMCU, internet connection is gathered by the NodeMCU by the system specified in the code

The channel id and API key is given from the ThinSspeak website created for the particular analysis. The output is viewed on the ThingSpeak account.

When there is a presence of light, the necessary action is taken.

CHAPTER 5

5. CIRCUIT DIAGRAM



CHAPTER 6

6.1 LITERATURE SURVEY

Travelling while owning a pet was always an obstacle for pet lovers so we wanted to search for a cheap solution for the same. With the help of circuit digest's magazine we found a circuit which would make feeding a pet easy in the absence of a keeper.

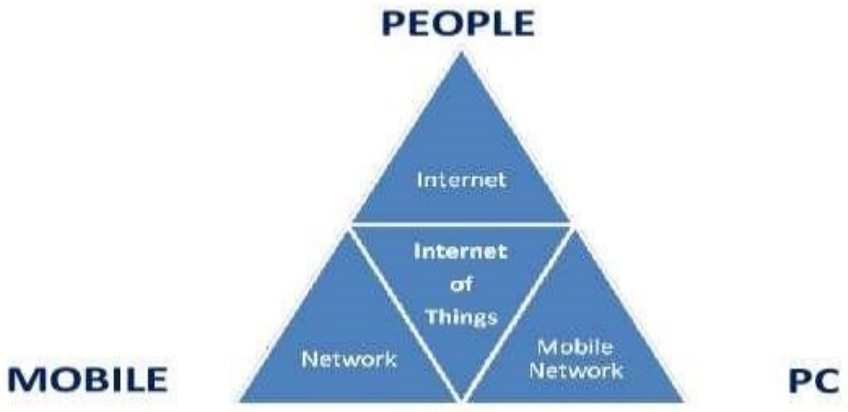
This circuit required an audrino uno coded with a simple code which moves a motor when the real time equals the feeding time of the pet. With the help of this the owner can set the feeding time of the pet and not have to worry about depending on neighbors or keepers again .

Many effective shadow detection algorithms have been proposed. A novel approach for shadow detection and classification is discussed in . The method uses invariant color models to identify and classify shadows. Shadow candidate regions are extracted first and then the candidate pixels are classified as self-shadow points or as cast shadow points by using the invariant color features. Here invariance properties of some color transformations are exploited for shadow detection.

These transformations describe the color configuration of each image point disregarding shadings, shadows and highlights. Even if there is a change in the imaging conditions like viewing direction, illumination condition and object's surface orientation the transformations are invariant. The luminance properties of shadows are exploited on the edge map obtained by applying the Sobel operator on input image. The luminance values of shadow regions are smaller than those in the surrounding lit regions. Like this shadow regions are extracted. The classification is done as cast shadows, if they belong to the scene background or as self-shadow if they are part of an object.

In September 2018 support for a Lua Flash Store (LFS) was introduced. LFS allows Lua code and its associated constant data to be executed directly out of flash-memory; just as the firmware itself is executed. This now enables NodeMCU developers to create Lua applications with up to 256Kb Lua code and read-only constants executing out of flash. All of the RAM is available for read-write data!

The input will be given through a Web App which will be based on IoT platform. Through this Web App, besides the regular on or off operation, the intensity of different devices can be controlled. Also, depending on the environment and outdoor conditions, the devices will themselves change their working. The LDR (Light Dependent Resistor) sensor will detect for the outdoor light. If not found sufficient, the system will start the tube light. Besides all this, the system will also support manual operation. The IR sensor will sense some aspects of the surroundings. If a person enters the room, the IR sensor will automatically start the prescribed devices.



Internet of Things can be defined as the collection of two terms: one is Internet, which is defined as networks of networks which can connect billions of users with some standard internet protocols. Internet connect several different sectors and department while using different technologies. Several devices like mobile, personal systems and business

organizations are connected to Internet. The second term is Thing, this term is basically mean to these devices or objects which turn into intelligent objects. Moreover this it is also a part of all objects of this real world. If we want to define IOT then we can not define it precisely and concisely but Vermesan et al. defined the Internet of Things as simply an interaction between the physical and digital worlds. The digital world interacts with the physical world using a plethora of sensors and actuators .

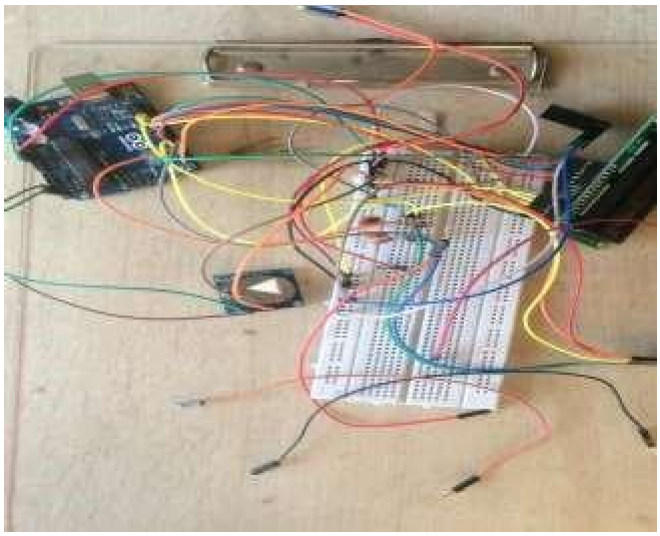
6.2 RESULT AND CONCLUSION

Automatic Pet Feeder works efficiently and fulfils the objective of feeding pet in absence of its master. It works for on house hold 230V AC supply. The servomotor rotates the Auger and food gets delivered into plate as programmed in the Arduino.

The design can be made more factory feasible and aesthetic. The auger can be made by 3D printing which can be

created as the auger is the moving part. Different materials can be used unlike acrylic sheet which was used in it. The APF could be made more advanced by installing cameras and audio box as it would make possible to interact with pet through smart phones as well as keep observation on it.

APF is indeed a helper to the owner for proper care taking of pet and more advancements can be expected in future.



CHAPTER 7

CODE FOR IOT PET FEEDER:

```
#include <DS3231.h>
#include <Servo.h>
#include <LiquidCrystal.h>
#include <Keypad.h>
```

```
const byte ROWS = 4; // Four rows
const byte COLS = 4; // Three columns
// Define the Keymap char
keys[ROWS][COLS] = {
  {'1','2','3','A'},
  {'4','5','6','B'},
  {'7','8','9','C'},
  {'*','0','#','D'}
};
// Connect keypad ROW0, ROW1, ROW2 and ROW3 to these
Arduino pins.
byte rowPins[ROWS] = { 2, 3, 4, 5 };
// Connect keypad COL0, COL1 and COL2 to these Arduino pins.
byte colPins[COLS] = { 6, 7, 8, 9 };
// Create the Keypad
Keypad kpd = Keypad( makeKeymap(keys), rowPins, colPins, ROWS,
COLS );

DS3231 rtc(A4, A5);
Servo servo_test;  //initialize a servo object for the connected
servo
LiquidCrystal lcd(A0, A1, A2, 11, 12, 13); // Creates an LC object.
Parameters: (rs, enable, d4, d5, d6, d7)
 //int angle = 0;
// int potentio = A0;  // initialize the A0analog pin for
potentiometer
 int t1, t2, t3, t4, t5, t6;

boolean feed = true; // condition for alarm
```

```
char key;  int     r[6];

void                                                    setup()
{
 servo_test.attach(10);   // attach the signal pin of servo to pin9 of
arduino
 rtc.begin(); lcd.begin(16,2);
                                                servo_test.write(55);
                        pinMode(A0,                     OUTPUT);
                        pinMode(A1,                     OUTPUT);
                        pinMode(A2,                     OUTPUT);

 }
void                                                    loop()
 {
lcd.setCursor(0,0);
int     buttonPress; buttonPress = digitalRead(A3);
if      (buttonPress==1)  setFeedingTime();

//Serial.println(buttonPress);
 lcd.print("Time:                                                  ");
 String      t      =      ""; t =      rtc.getTimeStr();   t1 = t.charAt(0)-
48;
     t2     =      t.charAt(1)-48;  t3     =      t.charAt(3)-48;  t4     =
 t.charAt(4)-48;  t5     =      t.charAt(6)-48;  t6     =      t.charAt(7)-48;

 lcd.print(rtc.getTimeStr());
lcd.setCursor(0,1);
                    lcd.print("Date:     ");     lcd.print(rtc.getDateStr());
```

```
 if (t1==r[0] && t2==r[1] && t3==r[2] && t4==r[3]&& t5==r[5] &&
t6==r[6] && feed==true)
 {
  servo_test.write(100);              //command to rotate the servo to the
specified angle delay(400);
        servo_test.write(55);         feed=false;
 }
 }
void                                                    setFeedingTime()
{
        feed =      true;    int i=0;
lcd.clear();
                lcd.setCursor(0,0);     lcd.print("Set    feeding     Time");
               lcd.clear();         lcd.print("HH:MM");   lcd.setCursor(0,1);

                                              while(1){    key = kpd.getKey();
                                                    char                    j;
                                                          if(key!=NO_KEY){
                                                         lcd.setCursor(j,1);
                                                             lcd.print(key);
                                        r[i]    =       key-48;             i++;    j++;
                                        if                                  (j==2)
              {              lcd.print(":");     j++;
              }              delay(500);
  }
                           if      (key  ==     'D')   {key=0;     break;      }
                                                                               }
}
```