DESIGN AND IMPLEMENTATION OF AUTONOMOUS PILL DISPENSER WITH IOT BASED PRESCRIPTION ACCESS

A PROJECT REPORT

submitted by

ABHIJITH E M
Reg. No : **MAC15EE004**
ARJUN RAMU AMBAT
Reg. No : **MAC15EE036**
MOHAMED RISVAN V
Reg. No : **MAC15EE083**
NIDHUN K
Reg. No : **MAC15EE094**

to

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

of

Bachelor of Technology
in
Electrical and Electronics Engineering

Department of Electrical and Electronics Engineering

Mar Athanasius College of Engineering
Kothamangalam, Kerala, India 686 666

MAY 2019

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING
MAR ATHANASIUS COLLEGE OF ENGINEERING
KOTHAMANGALAM

CERTIFICATE

This is to certify that the report entitled **DESIGN AND IMPLEMENTATION OF AUTONOMOUS PILL DISPENSER WITH IOT BASED PRESCRIPTION ACCESS** submitted by **Mr.Abhijith E. M, Mr. Arjun Ramu Ambat, Mr. Mohamed Risvan V and Mr. Nidhun K** to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the Degree of Bachelor of Technology in Electrical & Electronics Engineering is a bonafide record of the project work carried out by them under our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

Dr.Siny Paul	**Prof.Acy M Kottalil**
Project guide	Head of the Department

ACKNOWLEDGEMENT

It is a great pleasure to acknowledge all those who have assisted and supported us for successfully completing our project.

First of all, we thank God Almighty for his blessings as it is only through his grace that we were able to complete our project successfully.

We are deeply indebted to Dr. Solly George, Principal, Mar Athanasius College of Engineering for her encouragement and support.

We express our deep sense of gratitude to Prof. Acy M Kottalil, Head of Electrical & Electronics Engineering Department.

We also extend our deep sense of gratitude to our Project Guide, our Project Co-ordinator and Faculty Advisor, Dr. Siny Paul, Professor, Electrical & Electronics Engineering Department for her constant support and immense contribution for the success of our project.

We whole - heartedly thank all our classmates, for their valuable suggestions and for the spirit of healthy competition that exists between us.

ABSTRACT

Individuals consuming several tablets always struggle to take the correct medicine at correct time. Even though medications are combating diseases their benefits are not realized because most patients do not consume their medicines as prescribed. The goal of this project is to design and fabricate a device which dispenses all the medicines to be consumed at a time as a single package and give the general instructions to the user. Each device will be assigned with a unique identification number which corresponds to a patient. With the help of an application program interface, digital copy of prescription will be shared with the device during consultation or purchase from medicine pharmacy. The purchased medicines are fed into the device by selecting the correct item. Thereafter the device will automatically allot each medicine to specific cartridges which are stacked inside the device. The stack design is in such a manner that the sorting mechanism can pick desired medicine from each cartridge. From the digital prescription, medicine to be dispensed is identified and packed into a single package. The upper part of the device will be having the rotating stack arrangement of medicine cartridges along with a single inlet to feed medicines. An attractive touch screen interface can be provided for interaction with the device, also smart phone connected to same network act as an interface. Below every cartridge there will be outlet through which medicine picker mechanism will pick required number of tablets from the required cartridge. The picked medicines are then collected into a plastic container. Power supply, battery and control unit will be in the bottom layer of the device. Atmega 328p-Au is the micro controller used, along with the Wi-Fi module, drives for different actuators and ports for other input output modules will be incorporated on the mother board. Servo motors are used for precise angular control at different parts of the design. The control board is designed in AutoCAD Eagle. Physical Structure design is completed using AutoCAD Fusion 360 and 3D printer in Fablab.

Contents

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Caring of the aged is of a serious concern in the developing countries. Family members are responsible for the care and management of the old. In the modern age it is difficult for family members to be available all the time to support the aged. Today, in our society most families are nuclear. Elderly would prefer to remain independent and their desire for independence in natural, but it is a worry for their children. Sometimes despite their best effort, the aged fail to remember to take their medication on time. Automatic Medicine dispenser is one such approach to help them take their medicines efficiently. As the cost of in-home medical care rises, it has become more and more incumbent among individuals to opt for a device that effectively takes care of their medications. The automatic medicine dispenser serves the purpose. Another aspect that the automatic pill dispenser serves is adherence. Adherence is how closely patients correctly follow prescription instructions. Many people have prescription medications in the form of pills and are non-adherent to these prescriptions. We want to improve the quality of life for all users, by preparing the medication regimen automatically to eliminate non-adherence cases with regard to pill-based medication. This project is an attempt to create something that is more useful than current products in the market, by minimizing the work that the user must do in order to regularly take their medications. The product will help those who have medications and vitamins that they need to take on a daily basis. Keeping track of taking the right pill at the right moment each day can become a challenging experience for the elderly, as it is not as easy as it could be for a younger person. This fact is easily explained when we understand that many of the abilities such as sight, memory or logical capabilities tend to decrease in a proportional way to age once human beings have entered old age, making it difficult for them to remember which pill to take at the correct time, remembering to take them or confusing one pill with another as the person may not be able to distinguish one from another. This problem will most surely be a cause for concern for the people surrounding the pill-taker, as not taking a pill at the correct time can cause severe problems (such as organ rejection in a patient with organ transplant or heart attack in patients suffering from grave heart conditions).

On the other hand, several problems related to the high amount of pills nowadays prescribed to patients are found in hospitals or in retirement homes. In these places one of the main jobs is to give out to its patient the correct pills. Managing, sorting and giving out the pills to each one of the patients can sometimes have a high chance of error, with a patient or resident receiving one or more incorrect pills. It can be quite difficult to remember what pills to take when and how much the correct dosage is. To overcome this difficulty, the pill sorter will be able to take in a full months worth of several pills, sort them into the correct dosage, and dispense them at the correct interval. This method decreases the opportunity for error and makes it easier for the end user. This problem has been tackled in the past. However, the solution that others have come up with tends to resemble an alarmed box, still requiring manual sorting by the user, or a bulky countertop box without a direct user interface, requiring a somewhat high technical knowledge to operate. Our solution will overcome both of these common pitfalls with an intuitive user interface so even those with limited computer and technical knowledge can fully utilize the system and a simple design that avoids oversized motors, actuators, and containers. The end product will be a low-cost solution to an everyday problem. Current pill dispensing machines involve the user preloading compartments with the correct combination of pills. These pill compartments then get dispensed at time intervals making the loading process far less of an obstacle. Also, it is our hope to make the scheduling process extremely intuitive even for new users. Those with memory disabilities will no longer have to constantly remember when they have to take certain medications. In addition, recent design projects implement pill dispensing machines with a limited number of compartments. The pill dispensing machine that we are proposing will automatically dispense the right combination of pills, and will be capable of accommodating more medications.

The major requirements of customers who use this unit are the performance, serviceability, reliability, cost and safety. The performance required are lightweight, ease of use for both the caretaker and the patient, good construction to avoid potential tampering and a good a display unit. Inputs to the machine include, the time and date, how many cartridges are currently loaded into the system, how many pills are in each cartridge, and when pills need to be dispensed. Outputs include the pills themselves and any warning or informative notifications. The notifications are intended to alert the user of the status of the machine. At any time the user should be able to access the user interface and have an understanding of what it is doing. In addition to informing the user of its regular processes, it will also display if there is some kind of internal error within the machine. The idea is that if something happens that is out of the machines control, the user is at least notified so the issue can be corrected. The major components of this medication dispenser are a microcontroller interfaced with an LED display, a motor controller, an alarm system, a multiple pill container and dispenser. The overall operation is to facilitate the user to set the timings to dispense multiple pills at required timings. The Alarm system is designed to provide two types of indications one by lighting an LED and the other by providing a beep sound. The user is required to select from the touch screen for the pill to dispense and also to reset the alarm. The second alarm is to indicate the optimal availability of the pills in the container to warn the user to refill the dispenser with the required quantity of pills. The major objective is to keep the device simple and cost efficient. Elderly population can benefit from this device as it avoids expensive in-home medical care. The main objective of this project will be to solve the aforementioned problems by designing and synthesize a tool which will enable the owner to track every pill to ingest in an easy and simple way requiring no training or complex learning from their side in order to operate the device. This device will be an intelligent pill dispenser. The pill dispenser will be designed to prevent errors in hospitals and retirement homes where many pills have to be given daily to each one of the patients, each patient owning a device will not only drastically reduce the chances of errors occurring but also well optimize and speed up work for the caretakers/nurses by allowing the device to take care of pill management for them and freeing the time slot usually dedicated to that. This device is intended to log the pill name, number of pills and hours at which each pill is actually taken versus the time it should have been taken. Nowadays there has been an increasing awareness as the number of pills prescribed to elder people, stating that so many pills may have negative effects on the patient's health. The pillbox's logs will help gather data concerning this matter. There is a need to ensure the device is wirelessly connected so device management (defining the hours when a particular pill must be taken, number of pills in each compartment, etc.) as well as possibility of emitting warnings to the owner's relatives or nurses if needed (such as the patient not taking pills). In essence, the device will have to be a wireless electronic apparatus, having special attention to make it very precise as an error could prove fatal.

CHAPTER 2

LITERATURE REVIEW

Automatic pill reminder for easy supervision is a model proposed by A. Jabeena, et al[1] in 2017 International Conference on Intelligent Sustainable Systems (ICISS) that uses a small box divided into multiple compartments, each having a lid to open and an IR sensor attached to it. The box is connected to a real time clock, a microcontroller device Arduino Uno which processes the activities and accordingly displays the pill details and time of intake on the LCD attached to the box and a GSM module which sends message to the family physician or members in case the pill is not taken. The box consists of several compartments each having a pill for a definite time of the day. An electronic real time clock, with factory predetermined time interval, is automatically activated in sync with the pill intake timings. The real time clock will start beating and as it reaches the stipulated time of pill-intake, the buzzer will go on and message will be displayed regarding which pill to take and time to take each pill. This basic model drew us to the framework of the automatic pill dispenser whereby changes are made so that manual feeding of medicines to be consumed at specific intervals is replaced by automatic sorting by the machine by making strong design changes.

The proposed system is based on Android Operating system which will remind the users to take medicines on time through notification and automatic alarm ringing system. Medication reminder and healthcare an android application by Deepti Ameta,et al[2] provides greater insights on developing such a platform. Input to the system is the information entered by the patient which includes date, time, medicine name, doctors name, etc. The output of the system focuses on Medication Adherence. Medication adherence usually refers to whether patients take their medications as prescribed (eg, twice daily), as well as whether they continue to take a prescribed medication. Medication nonadherence is a growing concern to clinicians, healthcare systems, and other stakeholders (eg, payers) because of mounting evidence that it is prevalent and associated with adverse outcomes and higher costs of care. After login the patient will be able to view the list of all the registered doctors with their names

, contact information , phone numbers , hospital/clinic address , the availability of doctor accordingly and all other information which the Doctor registers at the time of Signing into the system. They can see the dropdown view of the diseases and can directly navigate to the list of Doctors. It also shows the next appointment with the Doctor. This helps the patients to find the Doctors disease wise. The services help them to understand the system properly so that it becomes useful and productive. Various modules that could be considered are:

Set Alarm module: It helps in reminding about the medicines. User can add details of his dosage schedules. Using the date field one can enter the starting and ending dates between which he has to take medicines. The time field shows the time of dosage and on that time the alarm will get rung. The user can add the description of the medicine, including name, purpose and other related description. All the information will be saved in the database. This makes any time availability of the patients records. They can change the ringtone of the alarm from the ringtones stored in the devices.

Get Notification module: Once the alarm is set then the user gets the notification. The users can activate or deactivate this accordingly. If he does not require the notification he can turn off it. If he requires this system then a notification will be sent into his device. Again if he wants the notification in email form, he can select the Notification through Email Mode or if he requires it in a message format he can go with Notification through Message Mode.

Enhancing Healthcare using m-Care Box (Monitoring non-compliance of medication) by Aakash Bharadwaj S, et al[4] proposed an effective healthcare management by IoT. The data produced by the raspberry pi is to be dealt efficiently. If a tablet is taken out the medical box, the medicine counter is decremented and the information the information is updated in the cloud. This inversely reflects in the end user of the doctor, patient, and the pharmacist. In the patients app, the number of tablets will be decremented. If the counter value updated to the cloud is less than 10, then both doctor and pharmacist will be updated and notified. Further the doctor will decide whether the patient has to continue with the same dosage of medicine or a new set. Once the doctor decides, he updates the table. This update is reflected in the cloud database which is further updated to the patients and the pharmacys interface. The new medicines will be then delivered to the patients residence after placing order by the patient. The mobile application keeps all of the above end users updated and gives them a measure of synchronization. The data from the cloud is taken and connected to the mobile app using JSON. Once the data is acquired, the local table is created and updated. The updating of the local database is done by using SQLite database. Hence, the data is easily read, updated and deleted using SQL commands. The same technique is used to update and display on all devices used by the patient, doctor and the pharmacies. A common ID is provided to all the three to make the process completely device-independent.

The proposed system thus finds its technical support from the papers mentioned above and the autonomous pill dispenser could now smoothly be able to transit the pills through a set of roller mechanism controlled by DC motor via a slit whose gap is controlled by a servomotor. The dispenser finds its application platform in difference with the the model proposed by Deepti Ameta by implementing the interfacing program in Raspberry Pi. An attractive user interface is made by block programming using MIT app inventor which is android based. Blocks consist of one or more declarations and statements. A programming language that permits the creation of blocks, including blocks nested within other blocks, is called a block-structured programming language. Blocks are fundamental to structured programming, where control structures are formed from blocks. The user interface would be welcoming the user by showing the first screen having the clock, time for the next medicine to be taken, settings and take now option. User if intends could take the medicines at that time or if he needs to change the settings he could select the settings option which would navigate to the time settings page. In time settings page the user could edit the time of consumption, refill the medicines or edit and update the prescription. The latter option if selected would then navigate the user to the final page where he could add/remove a particular medicines or its timings. The insights are drawn from the papers that helped to shape out the driver and controller circuit. The structured messages contains data for different operations such as adjusting unit dosage picker opening size, dispense the desired number of tablets and feed in operations are passed from the interfacing program running in the raspberry pi. The data required to generate the structured message is obtained from the cloud server where the data will get updated in real time. In general the controller sends signal to the driver circuit for actuating the DC motor and servomotor from data received from IoT and feedback is sent to the DC motor when the IR sensor detects a passage of the pill. Together with the unit dosage picker, controller and driver circuit, user interface, the autonomous pill dispenser does the intended functions.

CHAPTER 3

DESIGN AND FABRICATION OF AUTONOMOUS PIL DISPENSER

3.1 UNIT DOSAGE PICKER

Unit dosage picker is an integral part of autonomous pill dispenser. It contains a servomotor, feeder unit and roller, DC motor unit. The function of the unit dosage picker is to accept the medicines manually and pass the medicines in required number to the subsequent stage. For adjusting the passage according to the pills, a servomotor is used which gets the required data from the controller and adjusts the gap of the lower portion of the funnel. Counting the number of pills are made possible by the DC motor which controls the roller set by limiting the amount of pill passage and IR transmitter and reciever counts the pill passed to give necessary feedback.

3.1.1 MOTOR AND FEEDER UNIT

The basic motor and feeder set consists of a funnel shaped feeder that is designed so as to dispense optimal amount of medicines to the packing unit. Medicines are loaded in bunches and different set of medicines are of different sizes and in order to dispense flawlessly precise control of angular position is a requisite. This compelled to the selection of a SG90 servomotor and this cannot be done without a sensor for position feedback. A pair of IR sensor each having a transmitter and receiver hence serves the purpose. The receiver is primarily a photo diode with a filter attached to it so as to precisely monitor the transit of each of the pills. The size of the pill is determined by the dispenser using the data from IoT whereby digital prescription is directed by the doctor. The servomotor then adjusts the opening of the feeder through a set of rack and pinion gear drive.

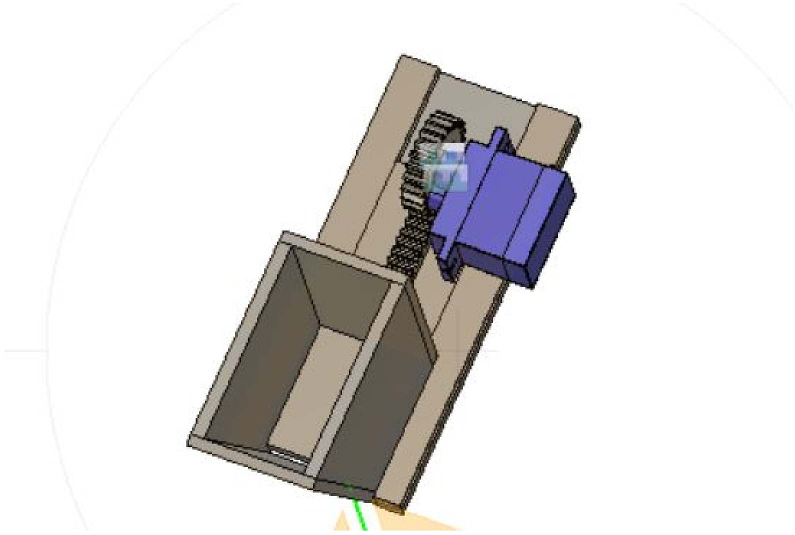

Figure 3.1: Motor and feeder unit

Servo motor(SG90)

A Servo Motor is a type of actuator that provides high precision control of linear or angular position.A simple servo motor consists of a small DC motor, a potentiometer for providing position feedback, a gear system for increased torque and a control system.As the motor rotates, the potentiometer's resistance changes, so the control circuit can precisely regulate how much movement there is and in which direction.

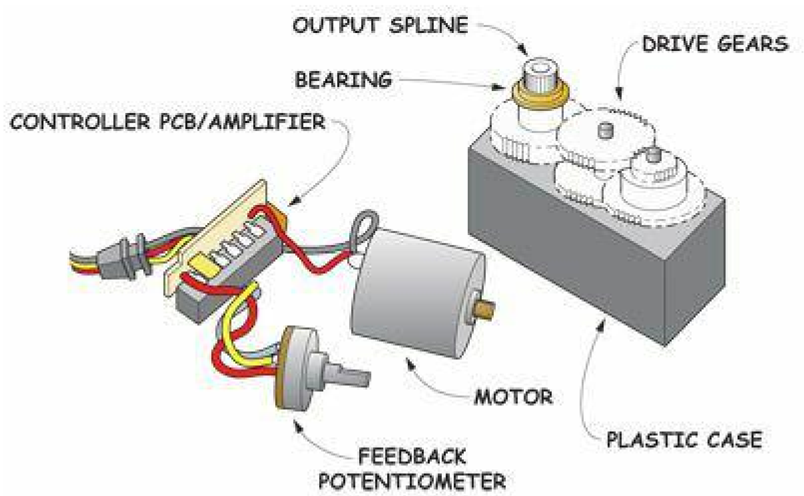

Figure 3.2: Servo motor

Servos are controlled by sending an electrical pulse of variable width, or pulse width modulation (PWM), through the control wire. There is a minimum pulse, a maximum pulse, and a repetition rate. A servo motor can usually only turn 90 in either direction for a total of 180 movement. The motor's neutral position is defined as the position where the servo has the same amount of potential rotation in the both the clockwise or counter-clockwise direction. The PWM sent to the motor determines position of the shaft, and based on the duration of the pulse sent via the control wire; the rotor will turn to the desired position. The servo motor expects to see a pulse every 20 milliseconds (ms) and the length of the pulse will determine how far the motor turns. For example, a 1.5 ms pulse will make the motor turn to the 90 position. Shorter than 1.5 ms moves it in the counter clockwise direction toward the 0 position, and any longer than 1.5 ms will turn the servo in a clockwise direction toward the 180 position. When the shaft of the motor is at the desired position, power supplied to the motor is stopped. If not, the motor is turned in the appropriate direction. The desired position is sent via electrical pulses through the signal wire. The motor's speed is proportional to the difference between its actual position and desired position. So if the motor is near the desired position, it will turn slowly, otherwise it will turn fast. This is called proportional control.There are two types of servo motors - AC and DC. AC servo can handle higher current surges and tend to be used in industrial machinery. DC servos are not designed for high current surges and are usually better suited for smaller applications. Generally speaking, DC motors are less expensive than their AC counterparts. When these servos are commanded to move, they will move to the position and hold that position. If an external force pushes against the servo while the servo is holding a position, the servo will resist from moving out of that position. The maximum amount of force the servo can exert is called the torque rating of the servo. Servos will not hold their position forever though; the position pulse must be repeated to instruct the servo to stay in position.

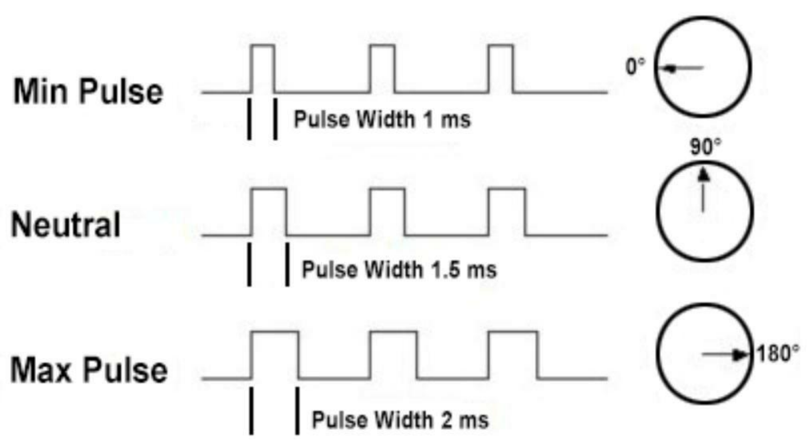

Figure 3.3: PWM timing control

3.1.2 ROLLER AND MOTOR UNIT

While the SG90 servomotor is used to control the parallel transit of pills to the packing unit, a DC motor is secured in a manner to control the series transit of pills whereby after one pill is dispensed, the DC motor reverses the direction of the roller to stop the forthcoming pill and thereby precisely measuring and controlling the passage of medicines. A DC motor is chosen as they are very rugged and easy to use and control is possible with commonly available and wide range of voltage levels. These motors can rotate in either direction and speed control is also possible. Also this motor is relatively cheap and handy compared to other motors. The motor used in the dispenser rotates at 100rpm and in order to control it to the required value of 6rpm a gear box is secured to the DC motor having a totality of 5 gears of which 3 gears handle the speed reduction in a 1:3 ratio and the other 2 two gears transmit power from the motor to the gear box and from the gear box to the roller.

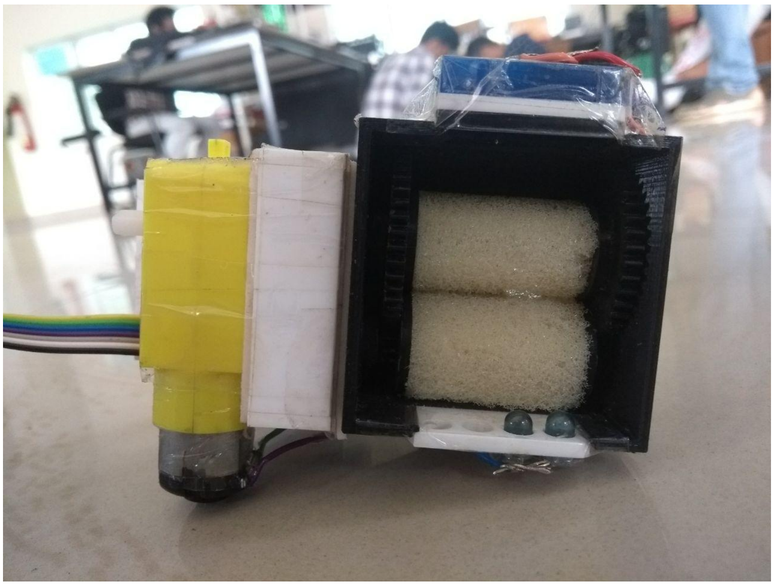

Figure 3.4: Roller and motor unit

IR transmitter and reciever

An infra-red sensor is an electronic device, that emits in order to sense some aspects of the surroundings. An IR sensor can measure the heat of an object as well as detects the motion.These types of sensors measures only infra-red radiation, rather than emitting it that is called as a passive IR sensor. Usually in the infra-red spectrum, all the objects radiate some form of thermal radiations. These types of radiations are invisible to our eyes, that can be detected by an infra-red sensor.The emitter is simply an IR LED (Light Emitting Diode) and the detector is simply an IR photo diode which is sensitive to IR light of the same wavelength as that emitted by the IR LED. When IR light falls on the photo diode, The resistances and these output voltages, change in proportion to the magnitude of the IR light received. The transmitter section includes an IR sensor, which transmits continuous IR rays to be received by an IR receiver module. An IR output terminal of the receiver varies depending upon its receiving of IR rays. Since this variation cannot be analysed as such, therefore this output can be fed to a comparator circuit. Here an operational amplifier (op-amp) of AD 8515 is used as comparator circuit.

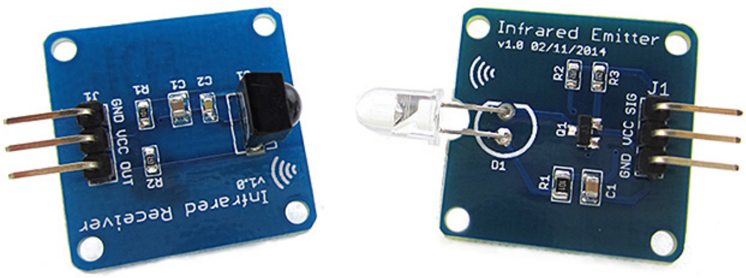

Figure 3.5: IR transmitter and reciever

When the IR receiver does not receive a signal, the potential at the inverting input goes higher than that non-inverting input of the comparator IC (AD8515). Thus the output of the comparator goes low, but the LED does not glow. When the IR receiver module receives signal to the potential at the inverting input goes low. Thus the output of the comparator (AD8515) goes high and the LED starts glowing. Resistor R1(100), R2(10k) and R3(330) are used to ensure that minimum 10 mA current passes through the IR LED Devices like Photo diode and normal LEDs respectively. Resistor VR1 (preset=10k) is used to set the sensitivity of the circuit Diagram.

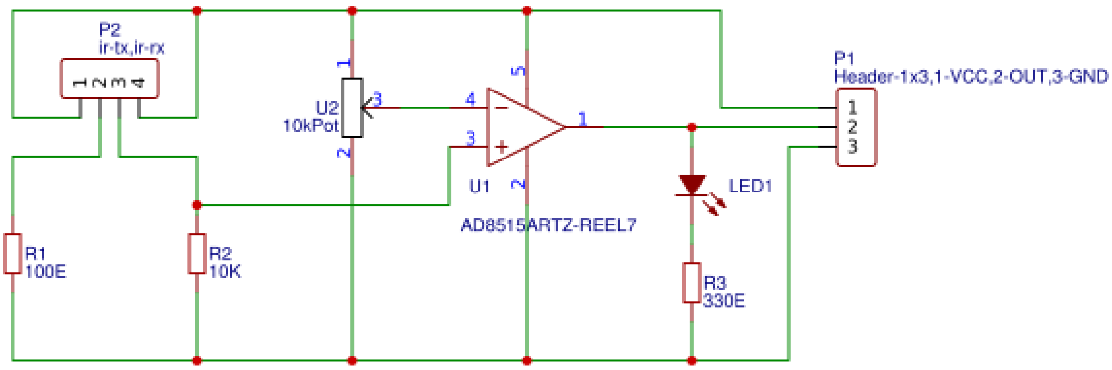

Figure 3.6: IR circuit diagram

Gear train

A series of gears are connected to form a gear train whose primary objective is the speed reduction and is connected between the DC motor and the roller unit such that the speed is reduced from 100 rpm to nearly 5 rpm which is in the ratio of 1:20. A set of 3 gears are used for speed reduction and one gear is used to transmit power from the DC motor shaft to gear train and one gear is used to transmit power from dc train to roller unit.

Figure 3.7: gear train

DC motor

RPM is one of the most important specifications of a DC motor. RPM, which stands for revolutions per minute, is the amount of times the shaft of a DC motor completes a full spin cycle per minute. A full spin cycle is when the shaft turns a full 360. The amount of 360 turns, or revolutions, a motor does in a minute is its RPM value. So a motor with an RPM of 24,000 is much more high speed than a motor which has 2400RPM. RPM is important when you need the motor to spin a certain number of times in a given time period. When speed is important, RPM is a crucial factor to look over when choosing a motor. In certain high-speed applications, it is imperative that motors that have high RPM are chosen. This may include applications such as washing machines with high-speed rinse cycles, treadmills that reach high speeds, and any such applications. Usually when the RPM value for a motor is specified, it normally is given with the voltage that will make it make that amount of turns per minute, such as 2400RPM @ 3V. Thus, the motor will spin 2400 times per minute when fed 3 volts DC into it. The no-load speed of a DC motor is the speed that the DC motor will turn when nothing is attached to its shaft. This is why it is called no load. The DC motor isn't loaded with an object. When a DC motor has nothing attached to its shaft, it is able to operate at its highest maximum speed. When it is then loaded with an object on its shaft, its speed will decrease. This is because it now has to bear with the weight of an object on it. The no-load speed serves as a reference to how fast the shaft of a motor will turn before weight is added to it. Thus, a circuit designer can have a frame of reference. The RPM value of a motor that will be specified will normally be the no-load speed. Stall torque is the torque produced on a motor when the output rotational speed is zero. This is why it is called the stall torque. It stalls the shaft of the motor, so that it longer spins and has rotational motion. This occurs when the torque load is either equal to or greater than the motor shaft torque. In this condition, the motor draws maximum current but does not rotate. Electric motors are devices which continue to provide torque when stalled. It's the same as a situation where a machinery is trying to lift up an object that is too heavy to lift. Though the object can't be lifted because it's too heavy, the machinery is still exerting force on it (to try to get it to lift up). This is the same with motors. If too much weight is applied to the shaft, more load than it can handle, the motor will not be able to sufficiently rotate. So stall torque has to do with the amount of weight a motor shaft can handle before it stalls. For this reason, stall torque is specified as gcm. This is the amount of grams of weight that a motor shaft can withstand per centimeter (cm) of its length. An example of a stall torque of a motor is 36 gcm. So the motor shaft has more than 36 gcm on its shaft, it will stall. Below this weight, it will have more torque than the load and, thus, will be able to rotate. Stall torque is a very important specification of a motor when weight on a shaft is important. If a circuit designer is only going to put a lightweight object on the motor's shaft, it isn't such a concern. If the designer is going to place a heavy item on the shaft, then it plays a huge role. The designer must be able to know whether that motor can handle that type of weight and still rotate. For motors spinning heavy objects, much more powerful motors must be used.

Figure 3.8: DC motor

3.1.3 Working

The unit dosage picker now gives the necessary feedback to the controller in the form of the pill count and provide the subsequent stages with the necessary amount of pills. Medicines are manually fed to the funnel shaped feeder and based on the data from

IoT, the devices senses the type and size of the pill and adjusts its gap accordingly for smooth passage. The DC motor controls the roller set so as to turn in one direction so that the pill is smoothly passed and reverses the direction so as to block the passage of subsequent pills for controlled transit. The pill count is given based on the data from the IR sensor having an IR transmitter and an IR receiver as feedback to the controller which actuates the DC motor.

3.2 Controller

In computing and especially in computer hardware, a controller is a chip, an expansion card, or a stand-alone device that interfaces with a peripheral device and controls the signal. The micro-controller is incorporated in the controller board. It also contains a FT232R FTDI driver,HC-05 bluetooth module, a reset switch, a crystal oscillator of 16 MHZ and a capacitor of 22 pf.

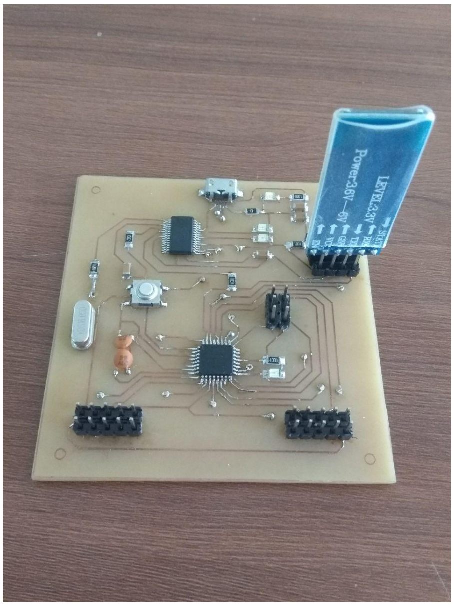

Figure 3.9: Controller

3.2.1 Microcontroller(ATMEGA328P-AU)

ATMEGA328P-AU is a high-performance, low-power consumption 8 bit microcontroller, and its core controller is AVR. It has 32KB of flash memory, and if the chip is power down, the data can still be retained. The architecture of the chip is advanced RISC having 131 Powerful Instructions Most Single Clock Cycle Execution, 32 x 8 General Purpose Working Registers, Fully Static Operation, Up to 20 MIPS Throughput at 20MHz, On-chip 2-cycle Multiplier. The number of digital-to-analog converters is eight and it has 23 input/output line. There are three types of chip interfaces: I2C, SPI and

USART. The chip has an internally calibrated oscillator and has six sleep modes: idle, ADC noise reduction, power save, shutdown, standby and standby expansion. Its special features include power-on reset and programmable brown-out detection, internal calibrated oscillator and external and internal interrupt sources. Its peripheral features include a two 8-bit timer/counters with separate pre-scaler and compare mode, one 16-bit timer/counter with separate pre-scaler, compare mode, and capture mode, a real time counter with separate oscillator, six PWM channels, 8-channel 10-bit ADC in TQFP and QFN/MLF package, a programmable serial USART, 6-channel 10-bit ADC in PDIP Package, programmable serial USART, master/slave SPI serial interface, byte-oriented 2-wire serial interface (Philips I2C compatible), programmable watchdog timer with separate on-chip oscillator, on-chip analog comparator and interrupt and wake-up on pin change. Its I/O and packages include 23 programmable I/O lines, 28-pin PDIP, 32-lead TQFP, 28-pad QFN/MLF and 32-pad QFN/MLF. Its operating voltage is in the range of 1.8 - 5.5V. Its temperature is in the range of -40deg.celsius to 85deg.celsius. Its power consumption at 1MHz, 1.8V, 25deg.celsius is given by 0.2mA in active mode, 0.1microA in power down mode, 0.75microA in power save mode and speed grade is given by 0 - 4MHz at 1.8 - 5.5V, 0 - 10MHz at 2.7 - 5.5.V and 0 - 20MHz at 4.5 - 5.5V. Its high endurance non-volatile memory segments consists of 4/8/16/32KBytes of in-system self-programmable flash program memory, 256/512/512/1KBytes EEPROM, 512/1K/1K/2K bytes internal SRAM, write/erase cycles of 10,000 Flash/100,000 EEPROM, data retention of 20 years at 85degC/100 years at 25degC, optional boot code section with independent lock bits having in-system programming by on-chip boot program and true read-while-write operation.

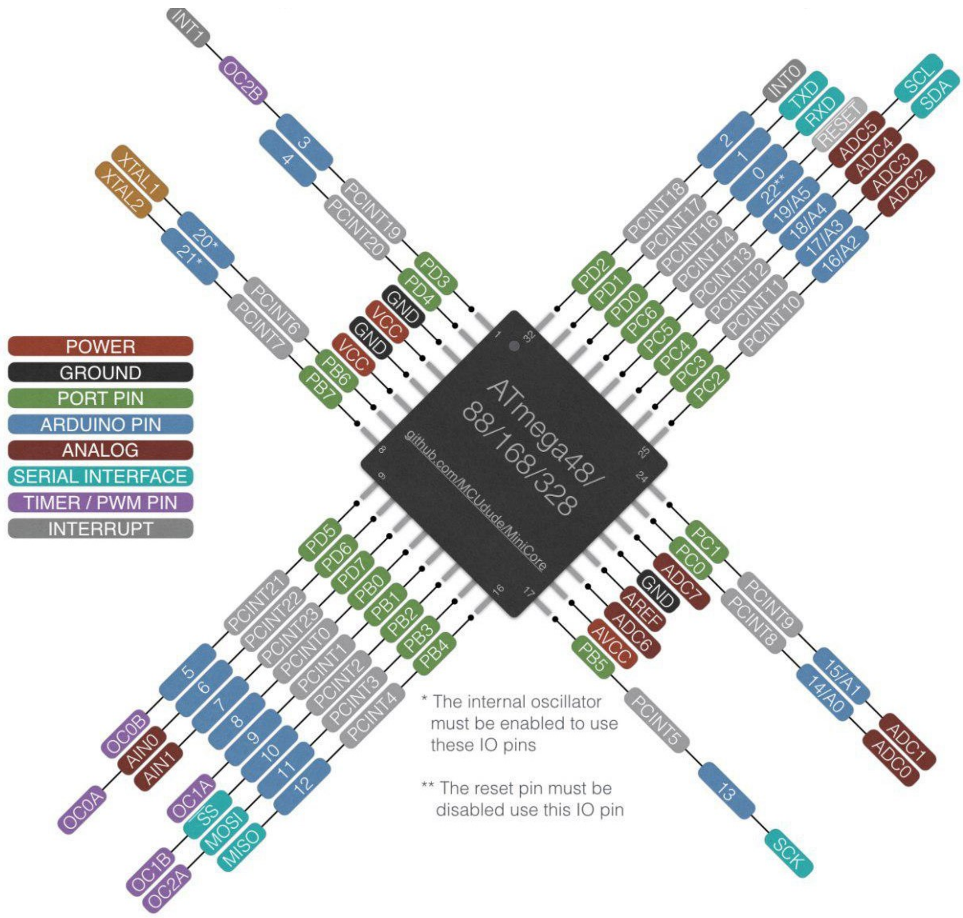

Figure 3.10: Pin diagram

VCC is the digital supply voltage and GND the ground. Port B (PB7:0) XTAL1/XTAL2/TOSC1/T is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tristated when a reset condition becomes active, even if the clock is not running. Depending on the clock selection fuse settings, PB6 can be used as input to the inverting Oscillator amplifier and input to the internal clock operating circuit. Depending on the clock selection fuse settings, PB7 can be used as output from the inverting Oscillator amplifier. If the Internal Calibrated RC Oscillator is used as chip clock source, PB7...6 is used as TOSC2...1 input for the asynchronous Timer/Counter2 if the AS2 bit in ASSR is set. Port C (PC5:0) is a 7-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The PC5...0 output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tristated when a reset condition becomes active, even if the clock is not running.

In PC6/RESET, if the RSTDISBL Fuse is programmed, PC6 is used as an I/O pin. The electrical characteristics of PC6 differ from those of the other pins of Port C. If the RSTDISBL Fuse is not programmed, PC6 is used as a Reset input. A low level on this pin for longer than the minimum pulse length will generate a Reset, even if the clock is not running. Shorter pulses are not guaranteed to generate a Reset. Port D (PD7:0) is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tristated when a reset condition becomes active, even if the clock is not running. AVCC is the supply voltage pin for the A/D Converter, PC3:0, and ADC7:6. It should be externally connected to VCC, even if the ADC is not used. If the ADC is used, it should be connected to VCC through a low-pass filter. PC6...4 use digital supply voltage, VCC. AREF is the analog reference pin for the A/D Converter. In the TQFP and QFN/MLF package, ADC7:6 serve as analog inputs to the A/D converter. These pins are powered from the analog supply and serve as 10-bit ADC channels.

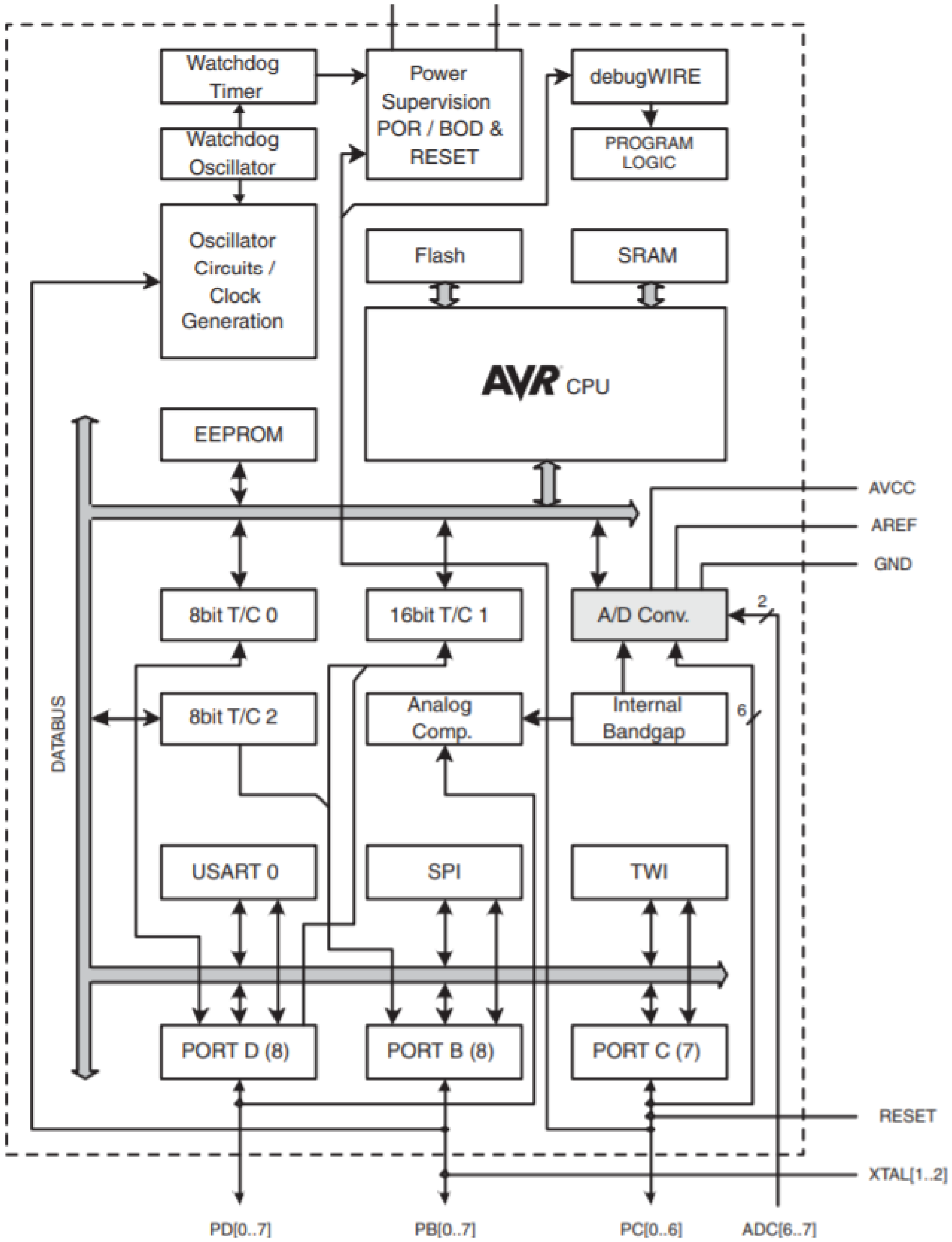

Figure 3.11: Internal block diagaram

The AVR core combines a rich instruction set with 32 general purpose working registers. All the 32 registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in one single instruction executed in one clock cycle. The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers. The ATmega328P-AU provides the following features:, 23 general purpose I/O lines, 32 general purpose working registers, three flexible Timer/Counters with compare modes, internal and external interrupts, a serial programmable USART, a byte-oriented 2-wire Serial Interface, an SPI serial port, a 6-channel 10-bit ADC (8 channels in TQFP and

QFN/MLF packages), a programmable Watchdog Timer with internal Oscillator, and five software selectable power saving modes. The idle mode stops the CPU while allowing the SRAM, Timer/Counters, USART, 2-wire Serial Interface, SPI port, and interrupt system to continue functioning. The Power-down mode saves the register contents but freezes the oscillator, disabling all other chip functions until the next interrupt or hardware reset. In Power-save mode, the asynchronous timer continues to run, allowing the user to maintain a timer base while the rest of the device is sleeping. The ADC Noise Reduction mode stops the CPU and all I/O modules except asynchronous timer and ADC, to minimize switching noise during ADC conversions. In Standby mode, the crystal/resonator Oscillator is running while the rest of the device is sleeping. This allows very fast start-up combined with low power consumption. Atmel offers the QTouch library for embedding capacitive touch buttons, sliders and wheels functionality into AVR microcontrollers. The patented charge-transfer signal acquisition offers robust sensing and includes fully denounced reporting of touch keys and includes Adjacent Key Suppression (AKS) technology for unambiguous detection of key events. The easy-to-use QTouch Suite toolchain allows you to explore, develop and debug your own touch applications. The device is manufactured using Atmels high density non-volatile memory technology. The On-chip ISP Flash allows the program memory to be reprogrammed In-System through an SPI serial interface, by a conventional non-volatile memory programmer, or by an On-chip Boot program running on the AVR core. The Boot program can use any interface to download the application program in the Application Flash memory. Software in the Boot Flash section will continue to run while the Application Flash section is updated, providing true ReadWhile-Write operation. By combining an 8-bit RISC CPU with In-System Self-Programmable Flash on a monolithic chip, the Atmel ATmega328P-AU is a powerful microcontroller that provides a highly flexible and cost effective solution to many embedded control applications. The ATmega328P-AU AVR is supported with a full suite of program and system development tools including: C Compilers, Macro Assemblers, Program Debugger/Simulators, In-Circuit Emulators, and Evaluation kits.

3.2.2 Bluetooth module(HC-05)

HC05 module is an easy to use Bluetooth SPP (Serial Port Protocol) module, designed for transparent wireless serial connection setup.. This serial port Bluetooth module is fully qualified Bluetooth V2.0+EDR (Enhanced Data Rate) 3Mbps Modulation with complete 2.4GHz radio transceiver and baseband. It uses CSR Blue core 04External single chip Bluetooth system with CMOS technology and with AFH (Adaptive Frequency Hopping Feature). The HC-05 Bluetooth Module can be used in a Master or Slave configuration, making it a great solution for wireless communication. By default the factory setting is SLAVE. The Role of the module (Master or Slave) can be configured only by AT COMMANDS. The slave modules cannot initiate a connection to another Bluetooth device, but can accept connections. Master module can initiate a connection to other devices. The user can use it simply for a serial port replacement to establish connection between MCU and GPS, PC to your embedded project,etc.

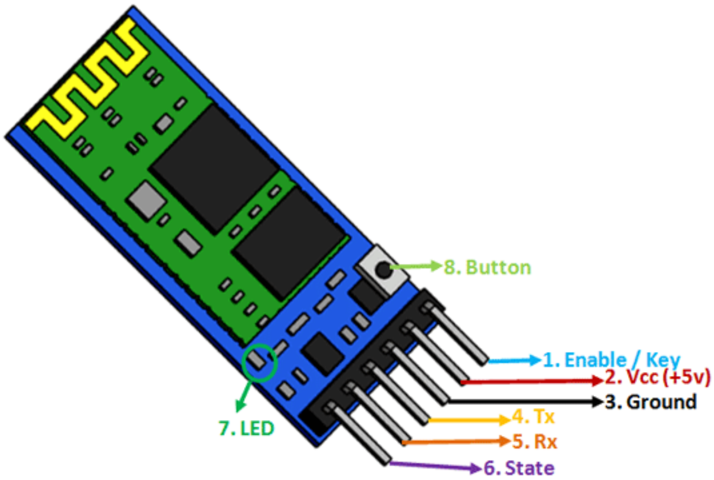

Figure 3.12: HC-05 Bluetooth module

The HC-05 Bluetooth Module has 6pins. They are as follows:

- ENABLE: When enable is pulled LOW, the module is disabled which means the module will not turn on and it fails to communicate .When enable is left open or connected to 3.3V, the module is enabled i.e. the module remains on and communication also takes place.
- VCC: Powers the module. Connect to Supply Voltage 3.3V to 5V
- GND: Ground pin, connects the system to the ground
- TRANSMITTER: Transmits Serial Data. Everything received via Bluetooth will be given out by this pin as serial data.
- RECEIVER: Receive Serial Data. Every serial data given to this pin will be broadcasted via bluetooth
- STATE: It acts as a status indicator. When the module is not connected to paired with any other Bluetooth device, signal goes Low. At this low state, the led flashes continuously which denotes that the module is not paired with other device. When this module is connected to/paired with any other Bluetooth device, the signal goes High. At this high state, the led blinks with a constant delay say for example 2s delay which indicates that the module is paired.

- BUTTON SWITCH: This is used to switch the module into AT command mode. To enable AT command mode, press the button switch for a second .With the help of AT commands ,the user can change the parameters of this module but only when the module is not paired with any other BT device. If the module is connected to any other Bluetooth device, it starts to communicate with that device and fails to work in AT command mode.

To communicate Smartphone with HC-05 Bluetooth module, Smartphone requires Bluetooth terminal application for transmitting and receiving data. HC-05 has red LED which indicates connection status, whether the Bluetooth is connected or not. Before connecting to HC-05 module this red LED blinks continuously in a periodic manner. When it gets connected to any other Bluetooth device, its blinking slows down to two seconds. So, when we want to communicate through Smartphone with HC-05 Bluetooth module, connect this HC-05 module to the PC via serial to USB converter by pairing HC-05 module to Smartphone.

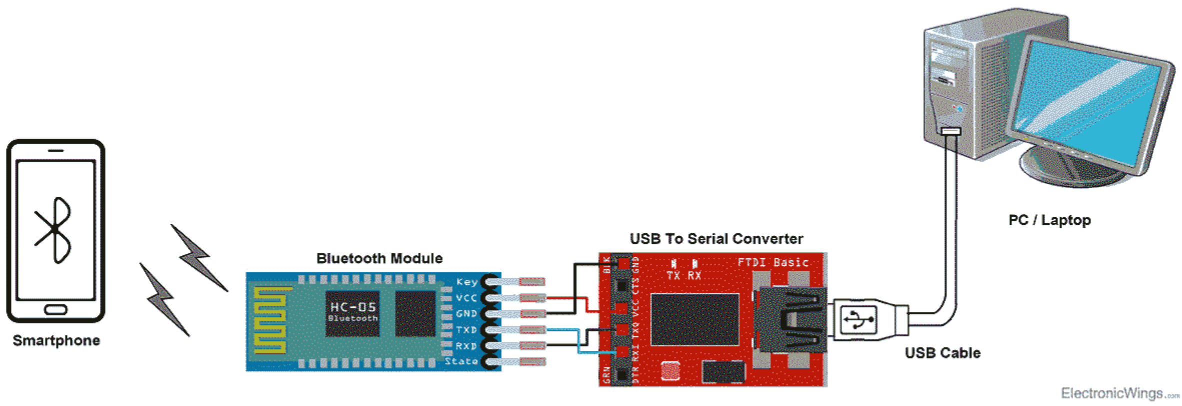

Figure 3.13: Bluetooth device connnection diagram

3.2.3 FT232R FTDI driver

Figure 3.14: FT232R FTDI driver

he FT232R is the latest device to be added to FTDIs range of USB UART interface Integrated Circuit Devices. The FT232R is a USB to serial UART interface with optional clock generator output, and the new FTDIChip-ID security dongle feature. In addition, asynchronous and synchronous bit bang interface modes are available. USB to serial designs using the FT232R have been further simplified by fully integrating the external EEPROM, clock circuit and USB resistors onto the device. The FT232R adds two new functions compared with its predecessors, effectively making it a "3-in-1" chip for some application areas. The internally generated clock (6MHz, 12MHz, 24MHz, and 48MHz) can be brought out of the device and used to drive a microcontroller or external logic. A unique number (the FTDIChip-ID) is burnt into the device during manufacture and is readable over USB, thus forming the basis of a security dongle which can be used to protect customer application software from being copied. The FT232R is available in Pb-free (RoHS compliant) compact 28-Lead SSOP and QFN-32 packages.

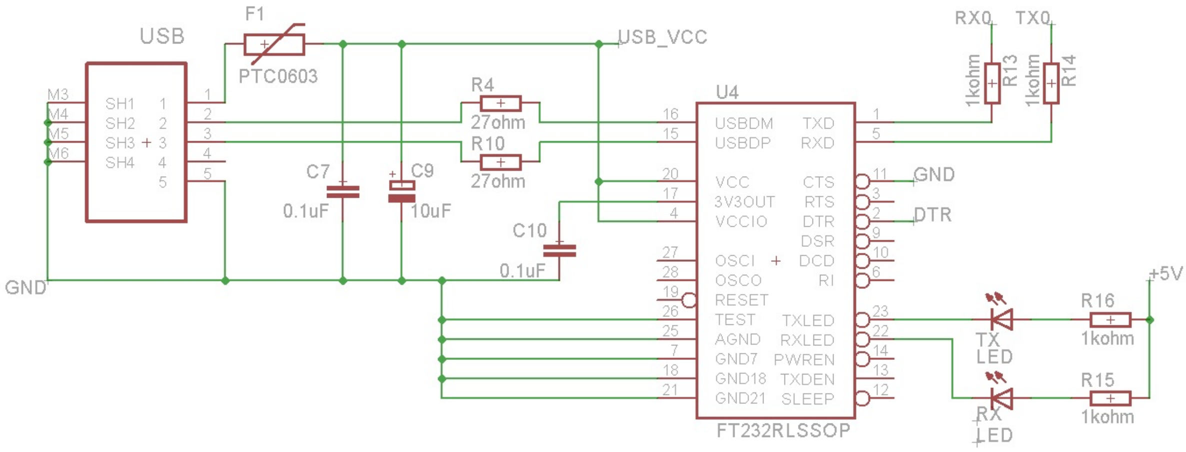

Figure 3.15: FT232R FTDI driver

It is a single chip USB to asynchronous serial data transfer interface, entire USB protocol handled on the chip - No USB-specific firmware programming required, has UART interface support for 7 or 8 data bits, 1 or 2 stop bits and odd / even / mark / space / no parity.It has fully assisted hardware or X-On / X-Off software handshaking. Data transfer rates ranges from 300 baud to 3 Megabaud (RS422 / RS485 and at TTL levels) and 300 baud to 1 Megabaud (RS232). It has In-built support for event characters and line break condition. It has auto transmit buffer control for RS485 applications.It transmits and receives LED drive signals. New 48MHz, 24MHz,12MHz, and 6MHz clock output signal Options for driving external MCU or FPGA. FIFO receive and transmit buffers for high data throughput. 256 Byte receive buffer and 128 Byte transmit buffer utilising buffer smoothing technology to allow for high data throughput. It has adjustable receive buffer timeout. It has Synchronous and asynchronous bit bang mode interface options with RD and WR strobes. It has Integrated 1024 bit internal EEPROM for I/O configuration and storing USB VID, PID, serial number and product description strings. It has device supplied preprogrammed with unique USB serial number. It has integrated 3.3V level converter for USB I/O. It has Integrated level converter on UART and CBUS for interfacing to 5V - 1.8V Logic. It has high I/O pin output drive option, integrated USB resistors, integrated power-on-reset circuit, fully integrated clock - no external crystal, oscillator, or resonator required, fully integrated AVCC supply filtering - No separate AVCC pin and no external R-C filter required, UART signal inversion option, USB bulk transfer mode, 3.3V to 5.25V Single Supply Operation, low operating and USB suspend current, low USB bandwidth consumption, UHCI / OHCI / EHCI host controller compatible, USB 2.0 Full Speed compatible, -40C to 85C extended operating temperature range, available in compact Pb-free 28 Pin SSOP and QFN-32 packages (both RoHS compliant).

3.3 Driver circuit

A driver is a circuit or component used to control another circuit or component. They are usually used to regulate current flowing through a circuit or to control other factors such as other components, some devices in the circuit. The term is often used, for example, for a specialized integrated circuit that controls high-power switches in switched-mode power converters. Driver circuit is used in this project to back the microcontroller since it provides only the signal and the action is facilitated by driver circuit.

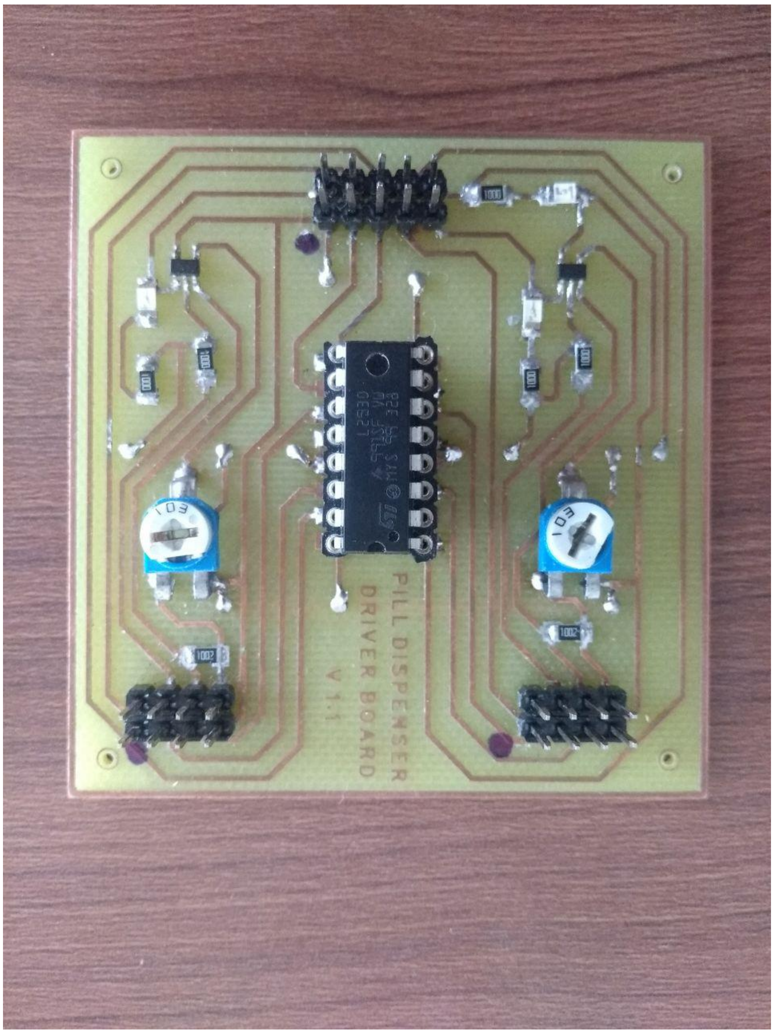

Figure 3.16: Driver circuit

3.3.1 L293D motor driver

Figure 3.17: L293D Motor driver

The L293D is a 16 pin IC, with eight pins, on each side, dedicated to the controlling of a motor. L293D is a typical Motor driver or Motor Driver IC which allows DC motor to drive on either direction. L293D is a 16-pin IC which can control a set of two DC motors simultaneously in any direction. It means that you can control two DC motor with a single L293D IC. It can drive small and quiet big motors as well. L293D Motor Driver IC works on the concept of H-bridge hence also known as Dual H-bridge Motor Driver integrated circuit (IC). H-bridge is a circuit which allows the voltage to be flown in either direction. As you know voltage need to change its direction for being able to rotate the motor in clockwise or anticlockwise direction, Hence H-bridge IC are ideal for driving a DC motor. In a single L293D chip there are two h-Bridge circuit inside the IC which can rotate two dc motor independently. Due its size it is very much used in robotic application for controlling DC motors.H-bridge is given this name because it can be modeled as four switches on the corners of H. The basic diagram of H-bridge is given below :

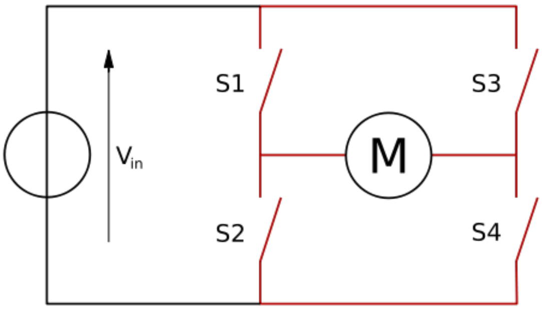

Figure 3.18: H-bridge

In the internal diagram shown below, when input 1 is closed and input 2 is opened transistor Q3 and Q2 are opened/operational, allowing current to flow from them. The motor connected across the output 1 and 2 now starts rotating. The motor direction of rotation is clock wise. One PNP and one NPN transistor is activated when one input is closed and other is opened.

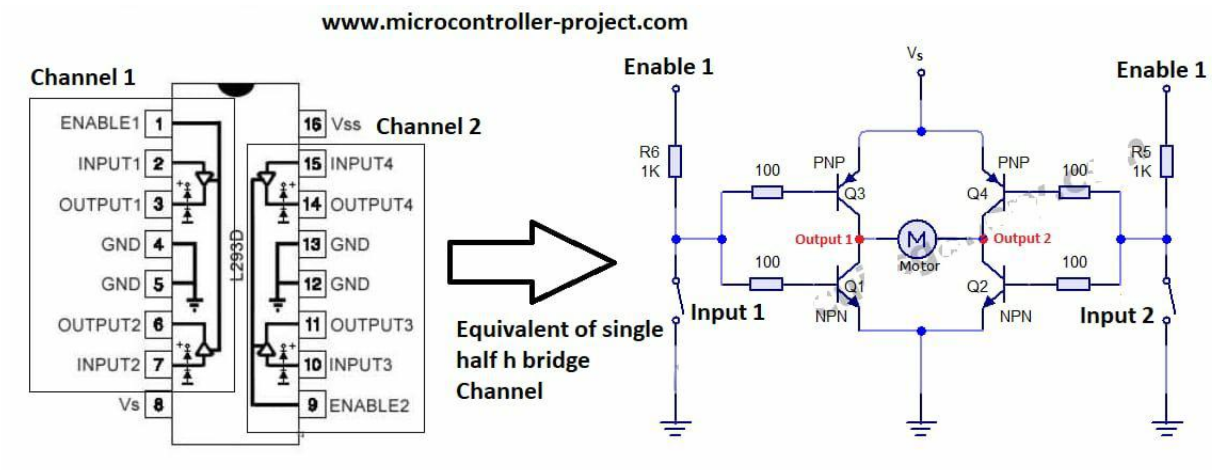

Figure 3.19: Internal diagram of L293D

In the pin diagram shown below, pin 1 signifies Enable 1-2, when this is HIGH the left part of the IC will work and when it is low the left part wont work. So, this is the Master Control pin for the left part of IC, pin 2 signifies INPUT 1, when this pin is HIGH the current will flow though output 1, pin 3 signifies OUTPUT 1, this pin should be connected to one of the terminal of motor, pin 4 signifies GND, ground pin, pin 5 signifies GND, ground pins, pin 6 signifies OUTPUT 2, this pin should be connected to one of the terminal of motor, pin 7 signifies INPUT 2, when this pin is HIGH the current will flow though output 2, pin 8 signifies VC, this is the voltage which will be supplied to the motor. So, if you are driving 12 V DC motors then make sure that this pin is supplied with 12 V, pin 9 signifies enable 3-4, when this is HIGH the right part of the IC will work and when it is low the right part wont work. So, this is the Master

Control pin for the right part of IC, pin 10 signifies INPUT 3, when this pin is HIGH the current will flow though output 3, pin 11 signifies OUTPUT 3, this pin should be connected to one of the terminal of motor, pin 12 signifies GND, ground pin, pin 13 signifies GND, ground pin, pin 14 signifies OUTPUT 4, this pin should be connected to one of the terminal of motor, pin 15 signifies INPUT 4, when this pin is HIGH the current will flow though output 4 and pin 16 signifies VSS, this is the power source to the IC. So, this pin should be supplied with 5 V.

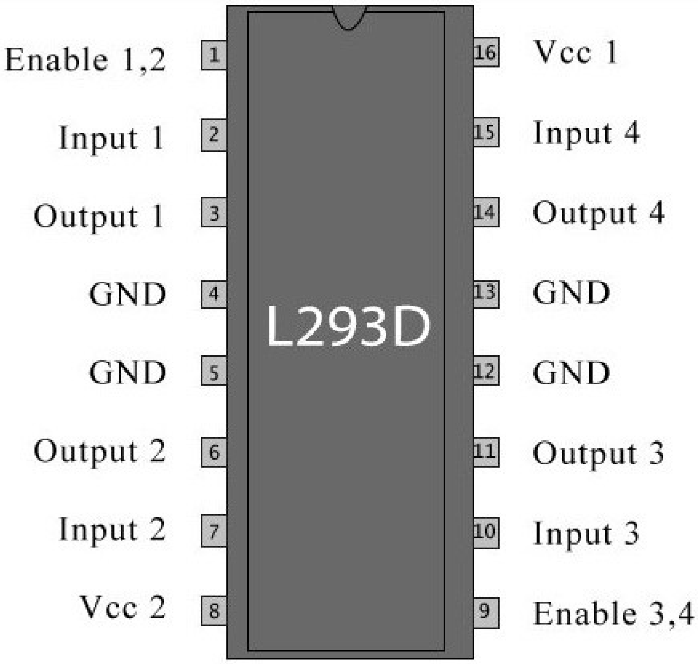

Figure 3.20: Pin diagram of L293D

The Motor Driver is a module for motors that allows to control the working speed and direction of two motors simultaneously . L293D provide bidirectional drive currents at voltages from 5 V to 36 V. The DC motor speed in general is directly proportional to the supply voltage, so if reduce the voltage from 9 volts to 4.5 volts then our speed become half of what it originally had. But in practice, for changing the speed of a dc motor we cannot go on changing the supply voltage all the time. The speed controller PWM for a DC motor works by varying the average voltage supplied to the motor

The input signals we given to PWM controller might be an analog or digital signal according to the design of the PWM controller. The PWM controller accepts the control signal and adjusts the duty cycle of the PWM signal according to the requirements. These diagram below shows the waveforms obtained as output at different voltage requirements. In these waves frequency is same but the ON and OFF times are different.

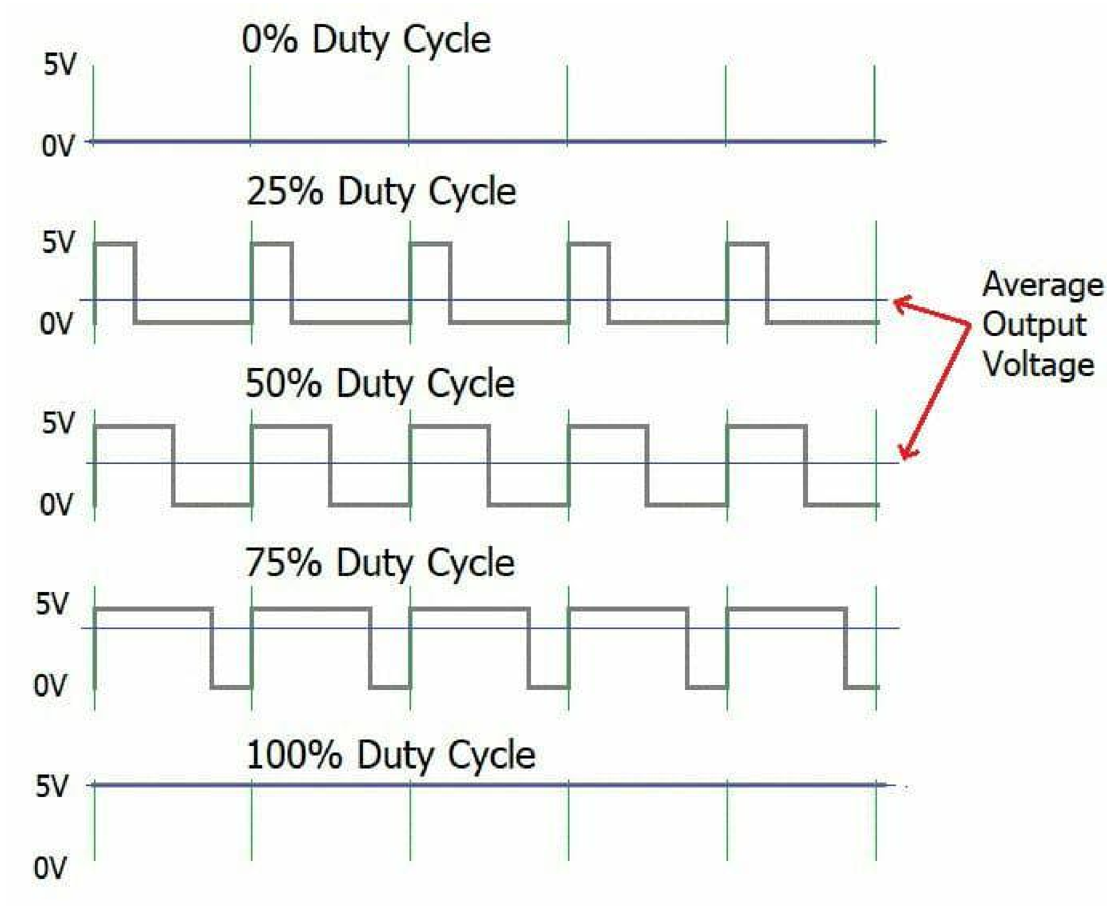

Figure 3.21: PWM DC motor speed control

The programming techniques are used to send appropriate control signals for machines or circuits in order to control. This can be achieved using various types of microprocessors and microcontrollers. We can control any circuit or machine by interfacing with atmega 328p microcontroller which is programmed with appropriate commands. Interfacing with atmega 328p microcontroller can be defined as transferring data from interfacing peripherals such as sensors, motors, machines, circuit components, and so on to atmega 328p microcontroller and vice versa. The main purpose of DC interfacing with atmega 328p microcontroller is for controlling the speed of the motor. The atmega 328p microcontroller block interfaced with input buttons and motor block with DC motor interfacing with atmega 328p microcontroller using motor driver as shown in the block diagram below.

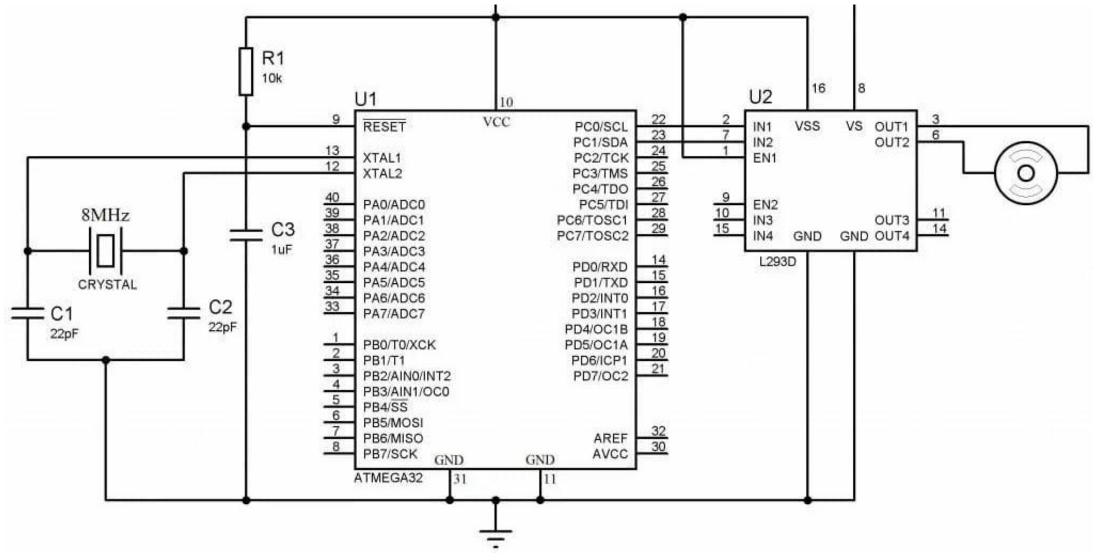

Figure 3.22: Interfacing with atmega 328p

The power supply provided to the experimental set-up is of 5V 5A using switch mode power supply. The device is also equiped with a battery back up of 12000mAh Li-ion battery for smooth function upto 24 hours.

The structured messages contains data for different operations such as adjusting unit dosage picker opening size, dispense the desired number of tablets and feed in operations are passed from the interfacing program running in the raspberry pi. The data required to generate the structured message is obtained from the cloud server where the data will get updated in real time. The controller executes operations according to the instructions received from the interfacing program. The different operations executed in the dispenser are adjusting unit dosage picker opening size and the dispense instruction. In adjusting unit dosage picker opening size, the instruction is given by "s openingsize1 openingsize2 openingsize3 openingsize4" whereby the first keyword is used to identify the operation corresponding to the specific instruction and "s" stands for set size operation followed by the four opening sizes corresponding to the tablet to be fed in. When the user selects a feed-in operation through the interface, medicine size and prescription details are fetched from the server and a feed in operation is initiated from a program running in raspberry pi. Set size instruction is sent to the controller through a Bluetooth communication module. In recognition of the set size, the instruction controller parses the received message and store in EEPROM of the microcontroller. Using a mapping function the size is converted (in mm) to the time corresponding to the duty-ratio of servo PWM generation. Corresponding to this, PWM is generated in the 7th pin in which the size controlling servo is attached. The dispense instruction is given by, "c count1 count2 count3 count4" where c stands for the count of the medicine to be dispensed followed by a count of medicines to be dispensed from each UDP unit. On recognition of dispense instruction by received message string comparison count data is fetched from the message and stored into an array. Controller enables the motor driver circuits in a sequential way. After enabling, the roller will start to roll backward for 1.5 seconds and then forward till an interrupt is received in the interrupt pin. An interrupt is generated when a tablet intersects the IR detector. During each detection, the count value in the array is decremented and the process is continued till the values become zero that I the required number of tablets are dispensed.

CHAPTER 4

EXPERIMENTAL SETUP WITH RESULT

4.1 Algorithm

The algorithm of the experimental set-up is made to start by reading the values from EPROM for servo motor settings. Values stored in the EPROM consists of the size and dimensions of the pill so that servomotor could adjust the gap of the funnel for easy transit of the pill. After reading the values from EPROM the PWM for servomotor is set. The next step is checking the availability of serial data from bluetooth module. This is the data from the Raspberry Pi to the controller via bluetooth module. If the data from the bluetooth module is available the message is read. The next step is parsing of this string to identify the message. After identifying the required message, the algorithm is switched into three cases. The first case is the size instruction which is read into EPROM and corresponding to this PWM for servomotor is set. The second case is the feed instruction based on which the user is informed of which medicines are to be placed in which containers present in the autonomous pill dispenser. The third case is the dispense instruction and the roller is rotated forward for the transit of pills until a positive signal is received from the IR sensor and after which the roller rotates backward. This cycle is then repeated.

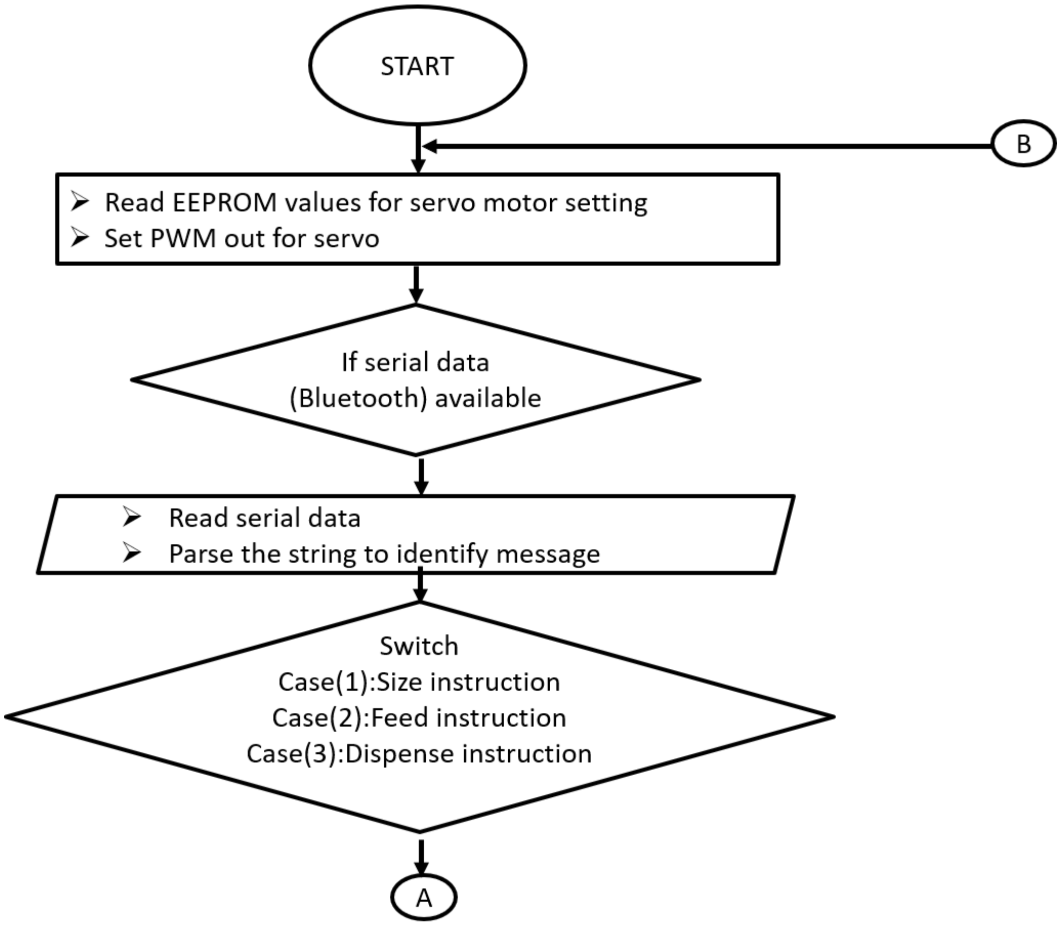

Figure 4.1: Code flow

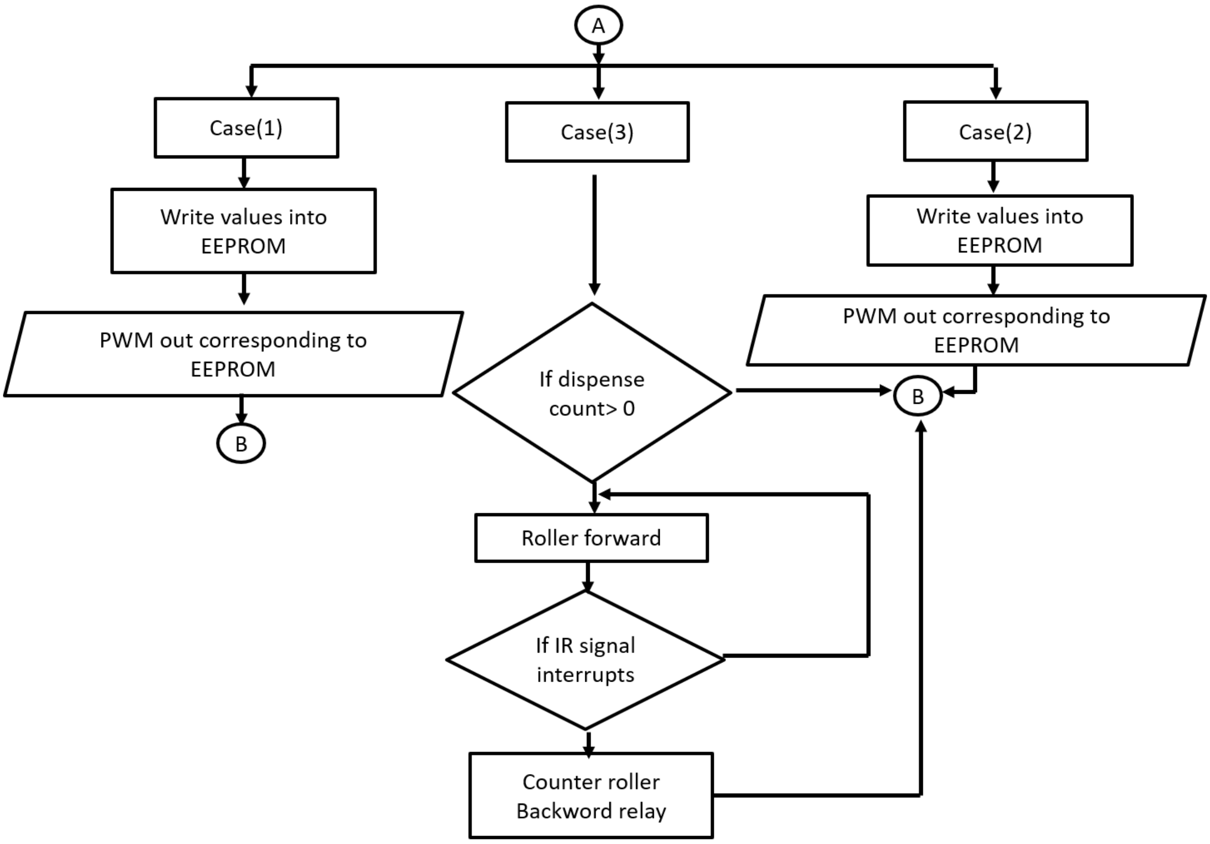

Figure 4.2: Code flow

4.2 Set-up

The experimental set-up contains the micro controller which is the heart of the system managing the overall operation of the system. It provides an attractive user interface and the keypad interface to interact with the external world. The text display provides information of the settings and also displays the present setting operation. The Program is written to help the user use the system effectively directing them to use the system without any problem.

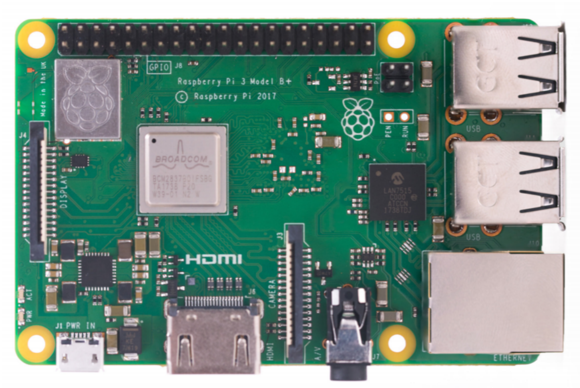

Figure 4.3: Raspberry pi

After the medicines are made input to the device the digital prescription is processed using raspberry pi 3 B+ which signals controller using bluetooth communicaton.The controller also gives feedback to raspberry pi. The data recieved serially through bluetooth communication which is standardised sequence of data, is processed and gives signal to driver. This data is used to actuate dc motors and servo motors. The driver board also contains proximity sensor which counts the transit of each pill and thereby directing the control board to do its intended functions.

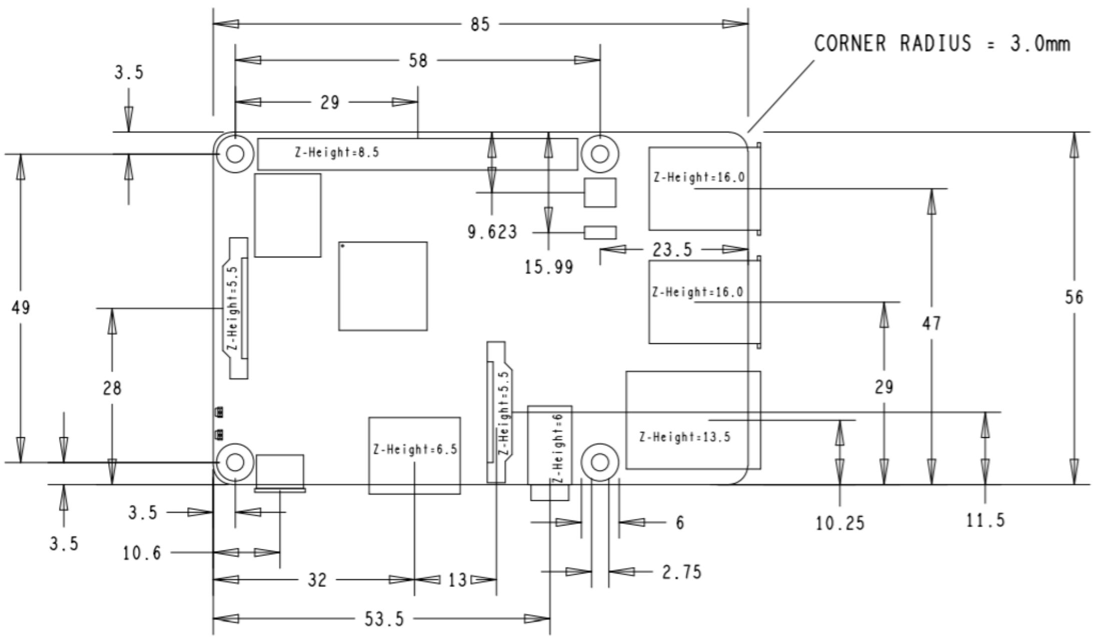

Figure 4.4: Raspberry pi

An attractive user interface is provided and is made by block programming using MIT app inventor which is android based. Block-based/graphical computer programming, using Scratch and related programming environments such as SNAP, as well as Lego's EV-3 programming environment, has become increasingly popular, as it makes programming structures much more intuitive. Combined with a variety of physical interfaces, such as the Makey-Makey, MakeBlock Arduino, or Scratch-X extensions, graphical programming can be used to control a wide variety of Arduino and other computer hardware. In computer programming, a block or code block is a lexical structure of source code which is grouped together. Blocks consist of one or more declarations and statements. A programming language that permits the creation of blocks, including blocks nested within other blocks, is called a block-structured programming language. Blocks are fundamental to structured programming, where control structures are formed from blocks.

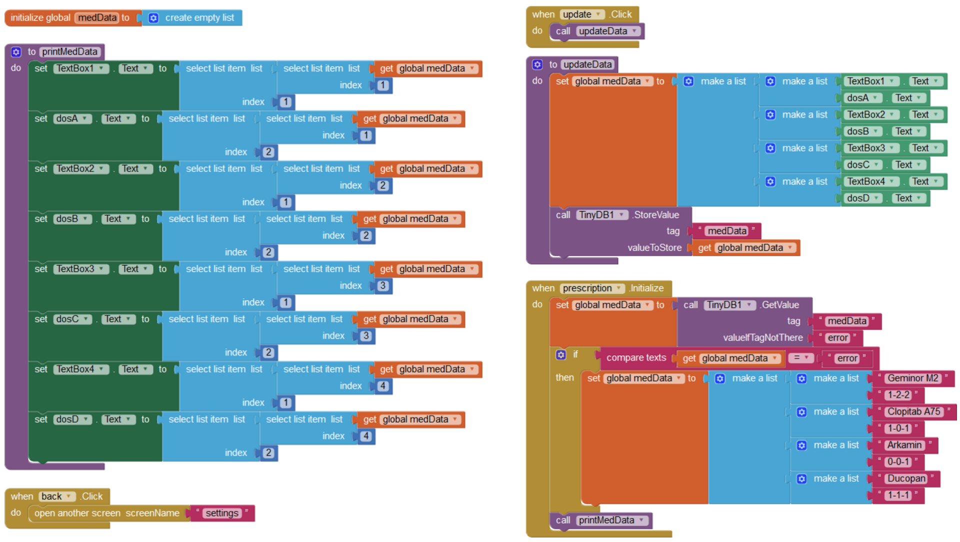

Figure 4.5: UI of upload page

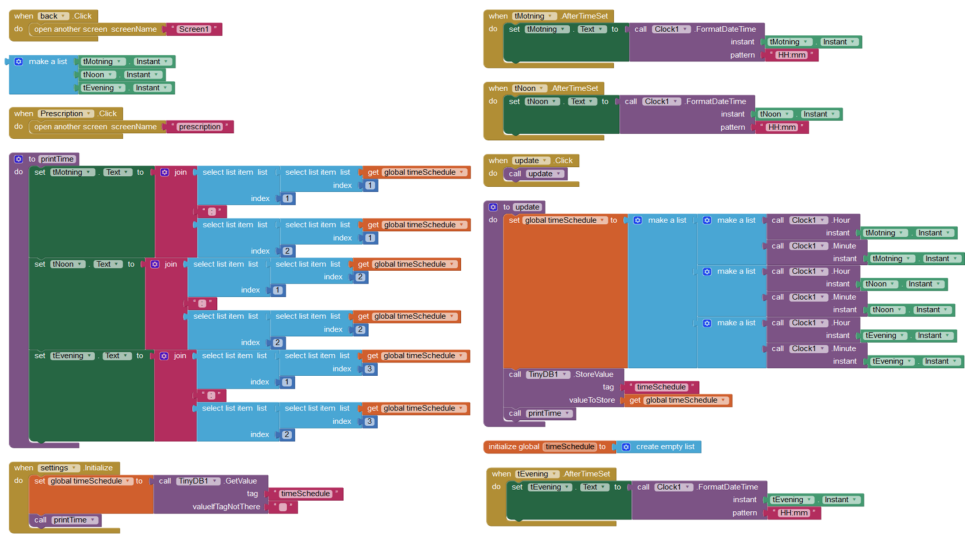

Figure 4.6: UI of settings page

The function of blocks in programming is to enable groups of statements to be treated as if they were one statement, and to narrow the lexical scope of objects such as variables, procedures and functions declared in a block so that they do not conflict with those having the same name used elsewhere. In a block-structured programming language, the objects named in outer blocks are visible inside inner blocks, unless they are masked by an object declared with the same name.

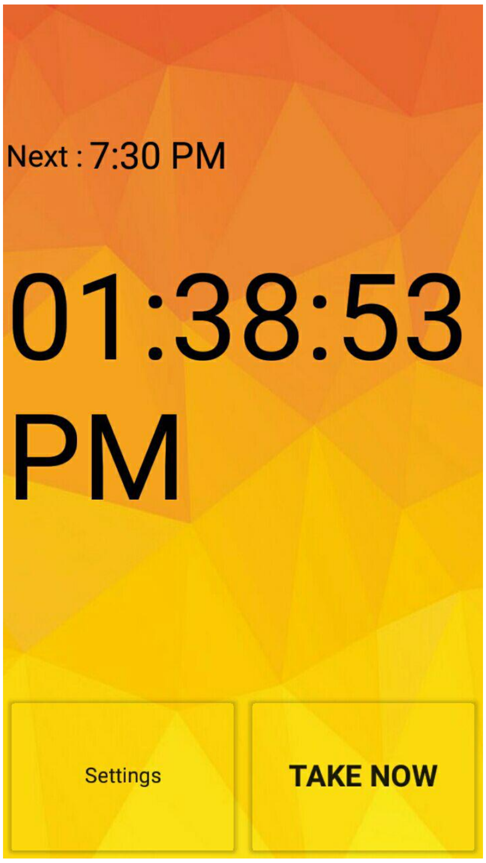

Figure 4.7: UI

This is the home screen of the user interface provided on the autonomous pill dispenser. In agreement with the convention and as use suggests, an attractive clock is being provided in large and time for the next medicine to be taken is shown. The navigation is provided to settings page which is shown in the next figure and to "take now" option. Take now is found useful when the user cant wait for the upcoming time for for having the medicine and he can take the medicines some time before as intended.

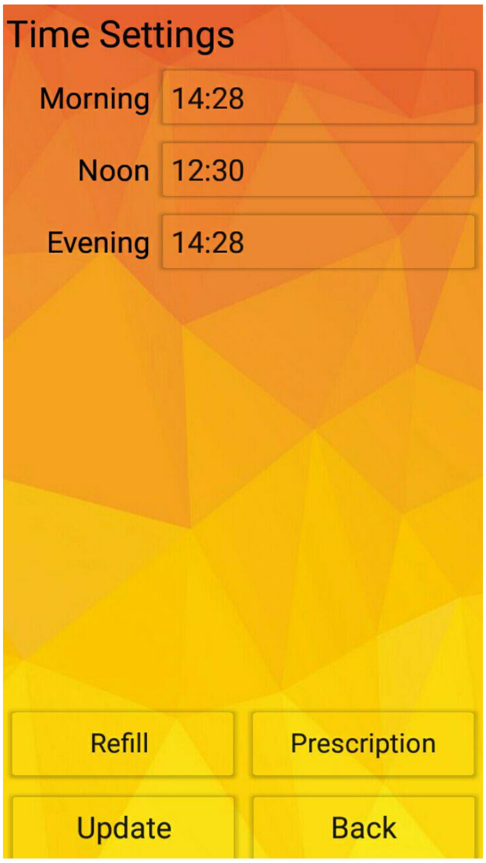

Figure 4.8: UI

The settings option in the home screen navigates to this page which is used primarily to set the time in which medicines should be taken respectively in morning, noon and evening. Apart from this it has option to refill, prescription access, update prescription and navigation to the home screen. Selecting the refill option would navigate into another screen whereby the user would be asked to feed the respective medicines to respective slots. Selecting the prescription would then navigate into the prescription page shown in the next figure whereby prescription could be edited and updated. Back option as mentioned above is used to navigate to the home screen.

Figure 4.9: UI

This is the final page of the user interface provided on the autonomous pill dispenser. This page is called the prescription page which displays the digital prescription and it displays the name of the medicines in sequential order with the dosage, that is how many medicines at each time. This is provided in the order of morning, noon and evening seperated by a hyphen. The two options provided in this screen are update and back. Update option is used to edit and update the prescription in fields like dosage, timing and adding or removal of medicines. Back option is used to navigate to the home screen.

The user interface would be welcoming the user by showing the first screen having the clock, time for the next medicine to be taken, settings and take now option. User if intends could take the medicines at that time or if he needs to change the settings he could select the settings option which would navigate to the time settings page. In time settings page the user could edit the time of consumption, refill the medicines or edit and update the prescription. The latter option if selected would then navigate the user to the final page where he could add/remove a particular medicines or its timings.

CHAPTER 5

Conclusion

An autonomous pill dispenser is successfully made with an application interface which dispenses medicines as per Medication Schedule Specification(MSS) either in packed or ready to consume manner. The device now timely reminds on dosages, monitor response towards reminders and adjust medication schedule and send notifications. It has the facility to send alarms four times a day. In case patient doesn't take medicines even after alarm them system sends a message to the particular number fed in the system memory. It is possible to program in order to change the number of times dispensing the medicines as per requirement.The user interface is made primarily by block programming using Massachusetts Institute of Technology app inventor and is android based. It would now address problems faced by elderly individuals with complex and prolonged medication schedule and patients with chronic diseases. The administration of medications now becomes easier and errors are substantially reduced. Also Full benefits will be realized due to timely medication. Its scope extends to adding a printing mechanism whereby patients could have their instructions printed on the package which would be particularly helpful while in transit. Another aspect to this autonomous pill dispenser is that when the medicine in the dispenser is exhausted the device could automatically alert its user and also it can opt for a fast delivery. Artificial intelligence and machine learning could also be implemented in this autonomous pill dispenser whereby the device could itself adjust the medication schedule with the help of available databases especially when the patient forgets to have the medicine on time. The device could also produce a database or a spreadsheet so that doctor could get a printed format of the past medication timings the patient had had the medicines.

References

1. A. Jabeena, Animesh Kumar Sahu, Rohit Roy and N. Sardar Basha, "Automatic pill reminder for easy supervision", ***International Conference on Intelligent Sustainable Systems (ICISS), November 17, 2017***

2. Deepti Ameta, Kalpana Mudaliar and Palak Patel "Medication reminder and healthcare an android application" ***International Journal of Managing Public Sector Information and Communication Technologies (IJMPICT), June 2, 2015***

3. Yuming Lu, Yingui Liu, Gengyu Li, Guozhi Song, Mingzhe Liu and Weixin Liu, " Design and application of an automatic packaging machine controller based on ATmega 128" ***The 2nd International Conference on Computer and Automation Engineering (ICCAE), 26-28 February, 2010***

4. S Aakash Bharadwaj, Divyank Yarravarapu, Sadiparala Charan Kumar Reddy, Thirumalaraju Prudhvi, K S P Sandeep, Obulam Siva Dheeraj Reddy "Enhancing Healthcare using m-Care Box" ***International Conference on Innovative Mechanisms for Industry Applications (ICIMIA), 21-23 February, 2017***