Chapter one

Introduction

1.1 General

Drugs can cure or control previously fatal diseases and help people live actively for decades longer. As people get older, the memory also is getting older too, and it's hard to remember everything and also body getting weaker, so a lot of people, especially elderly people, take a lot of medication on a daily basis to stay as healthy as possible. Some of that medication needs to be taken on exact time, or they don't function as well.

The technology revolution gives the medical filed the perfect tools to develop new systems based on microcontroller to prevent the medication administration error .The National Coordinating Council for Medication Error and Prevention defines a medication error as "... any preventable event that may cause or lead to inappropriate medication use or patient harm, while the medication is in the control of the health care professional, patient, or consumer. Such events may be related to professional practice, health care products, procedures, and systems including: prescribing; order communication; product labeling, packaging and nomenclature; compounding; dispensing; distribution administration; education; monitoring; and use"[1].there are several types of medication errors including:

1.1.1 Wrong route errors

Are situations where a medication is administered to the patient using a different route than was ordered or administration of a dose of medication that was never ordered for that patient.

1.1.2 wrong time errors

Are situations where the administration of a dose more than specific minutes depending on the site before or after the scheduled administration time [2].

1.2 Problem Statement

Most often people regardless of whether they are old or young tend to forget their medicines. Timely medication is very necessary for the cure of any disease. 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. As the cost of in-home medical care rises, it has become more important for individuals in need of supervised medical care to find a means in which to lower their medical care cost. So people need to find the best way to use assistive technologies to help them.

1.3 Objectives

The main aims of this system are:

- Design of an automatic medication dispenser using microcontroller.
- Simulation of this system
- Implementation system hardware.

1.4 Methodology

- An automatic medication dispenser is programmable device that allows nurses or users to specify the pill type and time to take pills. When it is time for patient medication, the machine generates sounds and light alarm and the motor rotates to take patient pills after that it goes back to "sleep" until the next alarm. The device contains four separate sub-boxes. Therefore, nurses or users can set information for four different pills.
- Atmega 32 microcontroller is selected to meet the required functionality of the device.
- The microcontroller is programmed using C language.
- The major components of this device are: Keypad, LCD display unit, speaker, motor and LEDs.

1.5 Thesis Layout

This thesis contains five chapters. Chapter One gives an introduction to the work, including general, problem statement, objectives and methodology. Chapter two presents literature review, previous work, introduction to microcontroller. Chapter Three discusses system description and component. Chapter Four illustrates system software and hardware considerations. Finally Chapter Five provides conclusion and recommendations.

.

Chapter Two

Literature Review and Theoretical Background

2.1 Automatic Medicine Dispenser

Automatic medicine dispenser is drug storage device or cabinet that electronically dispenses the right medicine at the right time. Also it reminds the client by voice and signal light so as to reduce medication errors, monitor, and control medication dispensing.

An automatic medication dispenser has great help to many individuals those who suffer from old age Alzheimer's and so on, so they tend to forgot taking their medicines. It also reduces the risk of incorrect medication or double dose. The Medication Dispenser system is critical to getting individuals the right dose, at the right time to keep people safe and independent as long as possible.

2.2 automatic medication dispenser types

The memory capacity decreases as human grows old. Remembering the dosages of medicines might become very difficult at times. Timely medication is very necessary for the cure of any disease but most often people tend to forget their medicines, so several products have attempted to solve this problem. In this review, the development of medication dispenser are considered. Abstract, advantages and disadvantages are stated for each one.

2.2.1 Pill tray dispenser

The pill tray shown in Figure 2.1, has a number of compartments that can be filled with medication. Each compartment can hold different sizes and type of medicines. The user is required to take the medicine from tray each day for a maximum of twenty eight days. It is systematic arrangement per day and time and user can easily check if medication is skipped so that it helps avoid double dosing. But this system is too reliant upon the user. Even if the medication is loaded correctly the user still has

to remember to take the medication, because it doesn't provide any alarm to indicate the time of taking the medicine.

2.2.2 Med-e-lert Medication Pill Dispenser

Figure 2.2 shows an automatic medication dispenser designed with twenty eight compartments, which meant to hold medication that's to be taken four times a day for seven days. It dispenses tablets at the right time and in the right quantity. At the pre-programmed times, the dispenser rotates, an alarm signal is heard, and the correct dosage comes into view through an opening in the lid. To get the medication the person needs to pick the unit up and turn it upside down. It is better than pill tray dispenser because it has an alarm which serves as a reminder that user should take his medication. The dispenser needs to refill at end of each week.

2.2.3 Philips medication dispenser

This product shown in Figure 2.3, contains the same feature of a med-E pill dispensing mechanism with an alarm to alert the client. The machine drops the correct pills in a small plastic container when the button is pushed by the user which also turns off the alarm The advantages of this device is that it alerts another individual of a missed dose rather than moving onto the next one. The device cost is very high so it is not suitable for poor individuals.

2.2.4 McLaughlin dispensing system

McLaughlin dispensing system includes a bedside dispenser, a programmable magnetic card, and a pharmacy computer. It a locked system that is loaded with the medications prescribed for a patient. At the appropriate dosing time, the bedside dispenser drawer unlocks automatically to allow a dose to be removed and administered. A light above the patient's door illuminates at the appropriate dosing time. Only certain medications fit in the compartmentalized cabinet [3].

Figure 2.1: Pill tray dispenser

Figure 2.2: Med-e-lert Medication Pill Dispenser

Figure 2.3: Philips medication dispenser

2.3 Literature Review

Ben Anderson and etal [4] design an automatic medication dispenser that allows the user to access the pills by entering a secret PIN code. This applied when loading a supply of pills or retrieving a dose of pills. The dispenser powered by a standard 120V 60 H.in case of emergency, the dispenser will be equipped with a back-up battery which will provide at least 12 hours of reserve power.

Kevin Villani and etal [5] design an automatic medication dispenser that has a cutting component and barcode reader the cutter is capable of cutting the pills into halves or quarters with significant accuracy. The barcode reader is included to minimize errors during the loading process. It works to reference an identification number that is given to each pill then uses this number to withdraw the information to that particular pill into the rest of the program.

Melody Moh and etal [6] design a system that tracks a patient's medicine supply using Radio Frequency Identification (RFID) technology and a scale. When new medicine is received, the patient assigns a uniquely identifiable RFID tag to its container, and inputs the corresponding schedule information into the system. The medicine storage device is then able to notify the patient when a particular medicine is to be taken. The scale is used to check that the patient actually takes the medicine.

2.4 Microcontroller

It is a highly integrated chip that contains all the components comprising a controller. It includes: CPU, RAM/ROM, I/O ports and timers. Microcontrollers sometimes called Embedded Microcontrollers, which just means that they are part of a larger device or system. Unlike a general purpose computer, which also includes all of these components, a microcontroller is designed for a specific task to control a particular system.

Microcontrollers are used in automatically controlled devices including power tools, toys, implantable medical devices, office machines, and engine control systems, printers, mobile phones, security systems, hearing aids, TV, radio, CD players and other types of embedded systems. All these products involve devices that require

some sort of intelligent control based on various inputs programmable microcontrollers contain general purpose input/output pins. The number of these pins varies depending on the microcontroller type. They can be configured to an input or an output state by software. When configured to an input state, these pins can be used to read external signals or sensors. When they are configured to the output state, they can drive external devices like LED displays and motors [7].

2.4.1 Types of microcontrollers

There are several different kinds of programmable microcontrollers at future electronics. This most common types categorized by several parameters including bits, flash size, RAM size, number of input/output lines, packaging type, supply voltage and speed. These parametric filters will allow user to refine his/her search results according to the required specifications .

- Classifications of microcontroller According to Number of Bits are:

i. A 8-bit microcontroller

Examples of 8-bit microcontrollers are Intel 8031/8051, PIC1x and Motorola MC68HC11 families.

ii. A 16-bit microcontroller

Examples of 16-bit microcontroller are 16-bit MCUs are extended 8051XA, PIC2x, Intel 8096 and Motorola MC68HC12 families.

iii. A 32-bit microcontroller

Examples of 32-bit microcontroller are Intel/Atmel 251 family, and PIC3x.

Future Electronics has a wide range of programmable microcontrollers, including PIC, low power, LCD, USB and wireless microcontrollers from several manufacturers.

AVR or Atmel AVR shown in Figure 2.5, is RISC based family of microcontrollers produced by Atmel Corporations. The acronym AVR has been reported to stand for advanced virtual RISC. The original AVR MCU was developed at a local ASIC house in Trondheim Norway, where the two founders of Atmel Norway were working as students. It was known as a µRISC (MicroRISC).

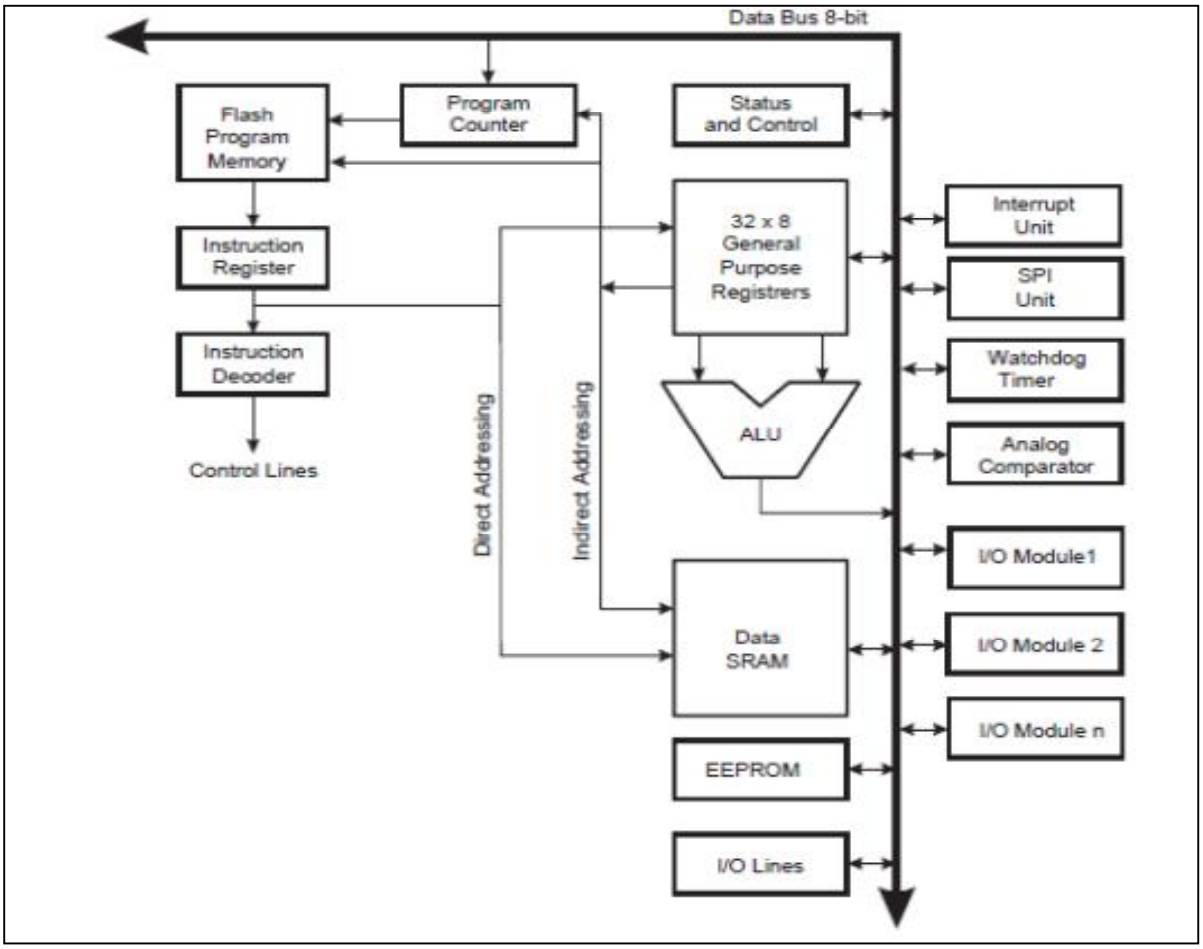


Figure 2.5**:** Block diagram of the AVR Architecture

Atmel says that the name AVR is not an acronym and does not stand for anything in particular but it may stand for the initials of the chip's designers: Alf and Vegard's RISC. The AVR is a Harvard architecture machine with programs and data stored separately. AVRs are generally classified into three broad groups:

i. Tiny AVRs:

- 1-8 KB program memory
- 8-20-pin package
- Limited peripheral set

ii. Mega AVRs:

- 4-256 KB program memory
- 28-100-pin package
- Extended instruction set and peripheral set

iii. Special purpose AVRs:

Mega AVRs with special features not found on the other members of the AVR family, such as LCD controller, USB controller, advanced PWM etc [7].

2.5 Stepper Motor

A stepper motor (or step motor) is a brushless, synchronous electric motor that can divide a full rotation into a large number of steps. Stepper motor can actually be used without any type of feedback loop. Since it motor moves in distinct steps as defined by steps angle. A stepper motor can be a good choice whenever controlled movement is required. They can be used to advantage in applications where you need to control rotation angle, speed, position and synchronism.

2.5.1 Stepper Motor Types:

There are three basic stepper motor types. They are:

• Variable-reluctance

• Permanent-magnet

• Hybrid

2.5.2 Fundamentals of operation

Stepper motors operate differently from direct current (DC) brush motors, which rotate when voltage is applied to their terminals. Stepper motors, on the other hand, effectively have multiple toothed electromagnets arranged around a central gear-shaped piece of iron. The electromagnets are energized by an external control circuit, such as a microcontroller. To make the motor shaft turn, first one electromagnet is

given power, which makes the gear's teeth magnetically attracted to the electromagnet's teeth. When the gear's teeth are thus aligned to the first electromagnet, they are slightly offset from the next electromagnet. So when the next electromagnet is turned on and the first is turned off, the gear rotates slightly to align with the next one, and from there the process is repeated. Each of those slight rotations is called a step, with an integer number of steps making a full rotation. In that way, the motor can be turned by a precise angle.

Chapter Three

System Description and Components

3.1 System Description

Figure 3.1 shows the proposed system block diagram of the proposed medication dispenser system. In this system the microcontroller is responsible for performing all functions and commands of the medicine dispenser. The components operate with interactive way, to receive information from user, the entered information is processed by the microcontroller chip, and then decides the correct action.

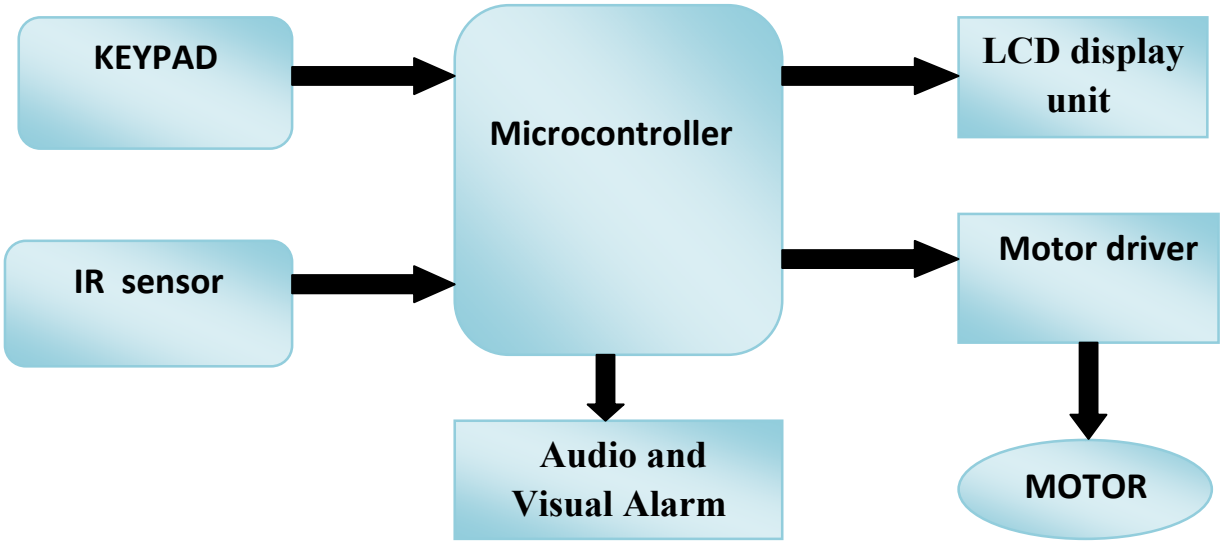


Figure 3.1: block diagram of medication dispenser system.

3.2 System Hardware Components

The major components of medication dispenser system are keypad, display unit, speaker, motor and LEDs.

3.2.1 Microcontroller (ATmega 32)

The microcontroller is the main part of the system. It is performing all functions and commands of the medicine dispenser. The microcontroller Atmega 32 is selected to

meet the required functionality of the device. The basic requirements of the microcontroller are a few input output ports to interface the keyboard, display unit, motor, and speaker.

To design the dispensing machine it is need a control system that continuously monitors the state of input devices and makes decisions based upon a custom program to control the state of output devices. The pin output, connection details are shown in appendix B.

3.3.2 Liquid-crystal display

Liquid-Crystal Display (LCD) shown in Figure 3.2, is a flat-panel display that uses the light-modulating properties of liquid crystals. LCDs are available to display arbitrary images (as in a general-purpose computer display) or fixed images with low information content, which can be displayed or hidden, such as preset words, digits, and 7-segment displays as in a digital clock.

The LCD screen is more energy-efficient and its low electrical power consumption, enables it to be used in battery-powered electronic equipment. It is an electronically modulated optical device made up of any number of segments controlling a layer of liquid crystals and arrayed in front of a light source (backlight) or reflector to produce images in color or colorless. LCDs are used in a wide range of applications including computer monitors, televisions, instrument panels, aircraft and signage. The pin output, connection details are shown in appendix B.

Figure 3.2: LCD display

3.3.3 Keypad 4x4

A keypad is a set of buttons arranged in a block or "pad" which usually bear digits, symbols and usually a complete set of alphabetical letters. If it mostly contains numbers then it can also be called a numeric keypad. Keypads are found on many alphanumeric keyboards and on other devices such as calculators, push- button telephones, combination locks, and digital door locks, which require mainly numeric input.

Figure 3.3: keypad 4x4

3.3.4 Stepper motor

A stepper motor is an electromechanical device which converts electrical pulses into discrete mechanical movements. The shaft or spindle of a stepper motor rotates in discrete step increments when electrical command pulses are applied to it in proper sequence. The motors rotation has several direct relationships to these applied Input pulses. The sequence of the applied pulses is directly related to the direction of motor shafts rotation. The speed of the motor shafts rotation is directly related to the frequency of the input pulses and the length of rotation is directly related to the number of input pulses applied. stepper motors have found their place in many different applications Some of these include printers, hard disk drives, medical equipment, fax machines, automotive and many more applications.

Figure 3.4: stepper motor

3.3.5 Motor driver (ULN2003)

Figure 3.5, shows ULN2003 driver. It is the actual device that energizes and de-energizes the circuit of the motor so that it can start or stop it .ULN takes logic inputs from the microcontroller and supply enough current to the stepper motor to meet maximum torque requirements. the pin out of the ULN2003 after being programmed for the system and all connection details show in appendix B.

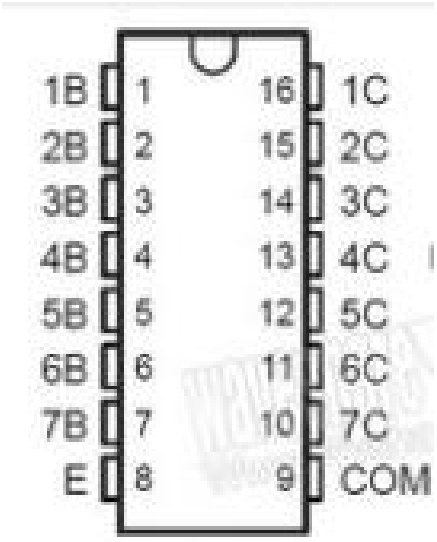


Figure 3.5: ULN2003

3.3.6 Infrared sensor

An infrared sensor is an electronic device that emits in order to sense some aspects of the surroundings and measure the IR light that is transmitted in the environment to find objects by an IR LED. An IR sensor consists of an emitter, detector and associated circuitry. The circuit required to make an IR sensor consists of two parts: the emitter circuit and the receiver circuit.

The emitter is simply an IR LED (Light Emitting Diode) and the detector is simply an IR photodiode which is sensitive to IR light of the same wavelength as that emitted by the IR LED. When IR light falls on the photodiode, its resistance and correspondingly, its output voltage, change in proportion to the magnitude of the IR light received. This is the underlying principle of working of the IR sensor. Advantages of infrared sensor are can detect infrared light over a large area, they can operate in real-time, the IR sensor uses non-visible light for detection and they are cheap sensors. But this sensor is very sensitive to IR lights and sunlight, and this is the main reason that an IR sensor is used with great precision in spaces with low light.

3.3.7 Audio and Visual Alarm

The Speaker is an audio alarm provided to give a beep sound to warn the patient regarding the time to take the Tablet.

LED (Light emitting diode display) is a visual alarm display simple red light to have the attention of the user and emergency indication.

Chapter four

System software and hardware consideration

4.1 System Software

System software of this study includes program code and simulation. The simulation results is illustrated along with the screen shots of the system. The system is simulated using Proteus simulation program.

4.1.1 Code

System code is written for the ATmega32 AVR microcontroller using C program language. It is chosen for ease of programming, and because the microcontroller contains enough program space to the point where code size does not become an issue on this project. The complete code is shown in appendix A. The program code is written using BASCOM-AVR program which is a very powerful and easy to use compiler for the AVR series of microcontrollers developed by atmel.

The program comes with a very user friendly interface and set of simple commands, and provides more flexibility than other programs .with some basic knowledge of c or c++ anybody can write successful program using bascom .

The best thing about BASCOM is that it can directly burn any flash file into a microcontroller using just four wires connected to computer's parallel port. The program's simulator, checker and emulator tools can help you scan your program for errors before burning it to any microcontroller.

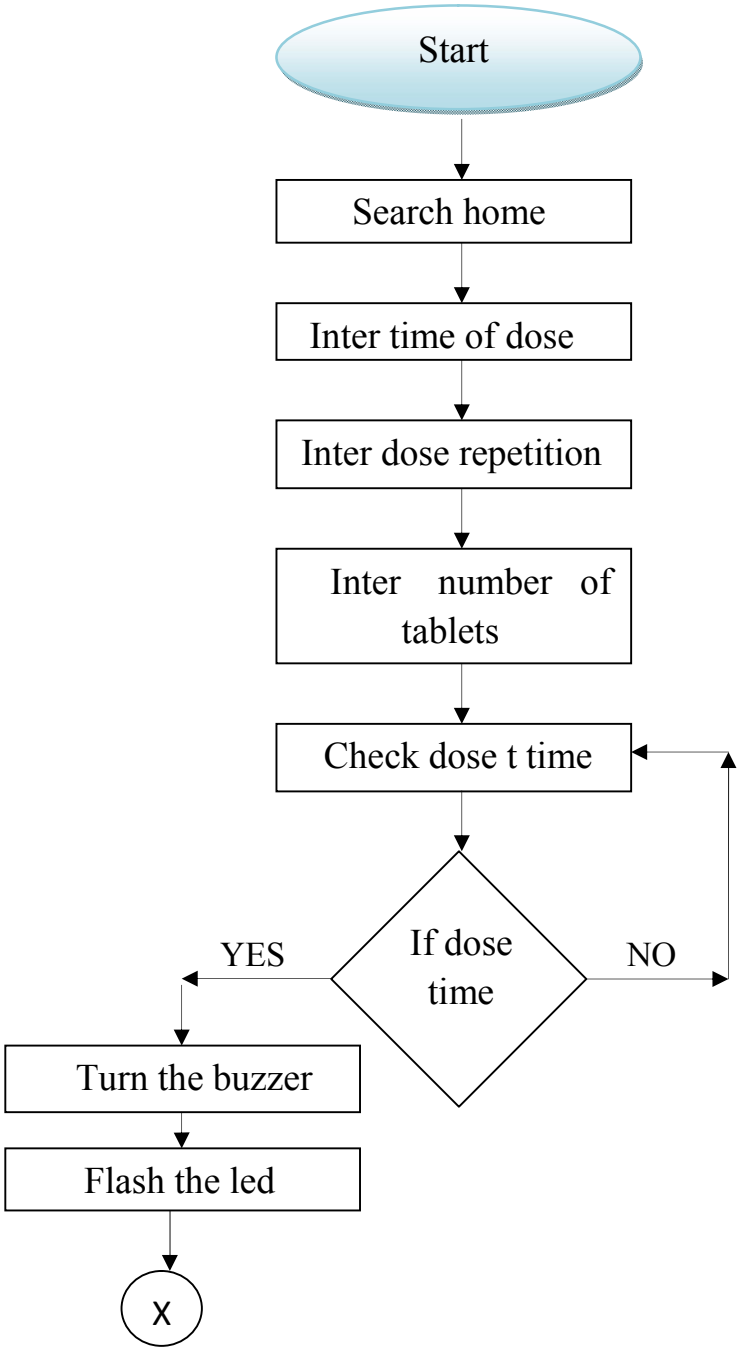
4.1.2 System simulation

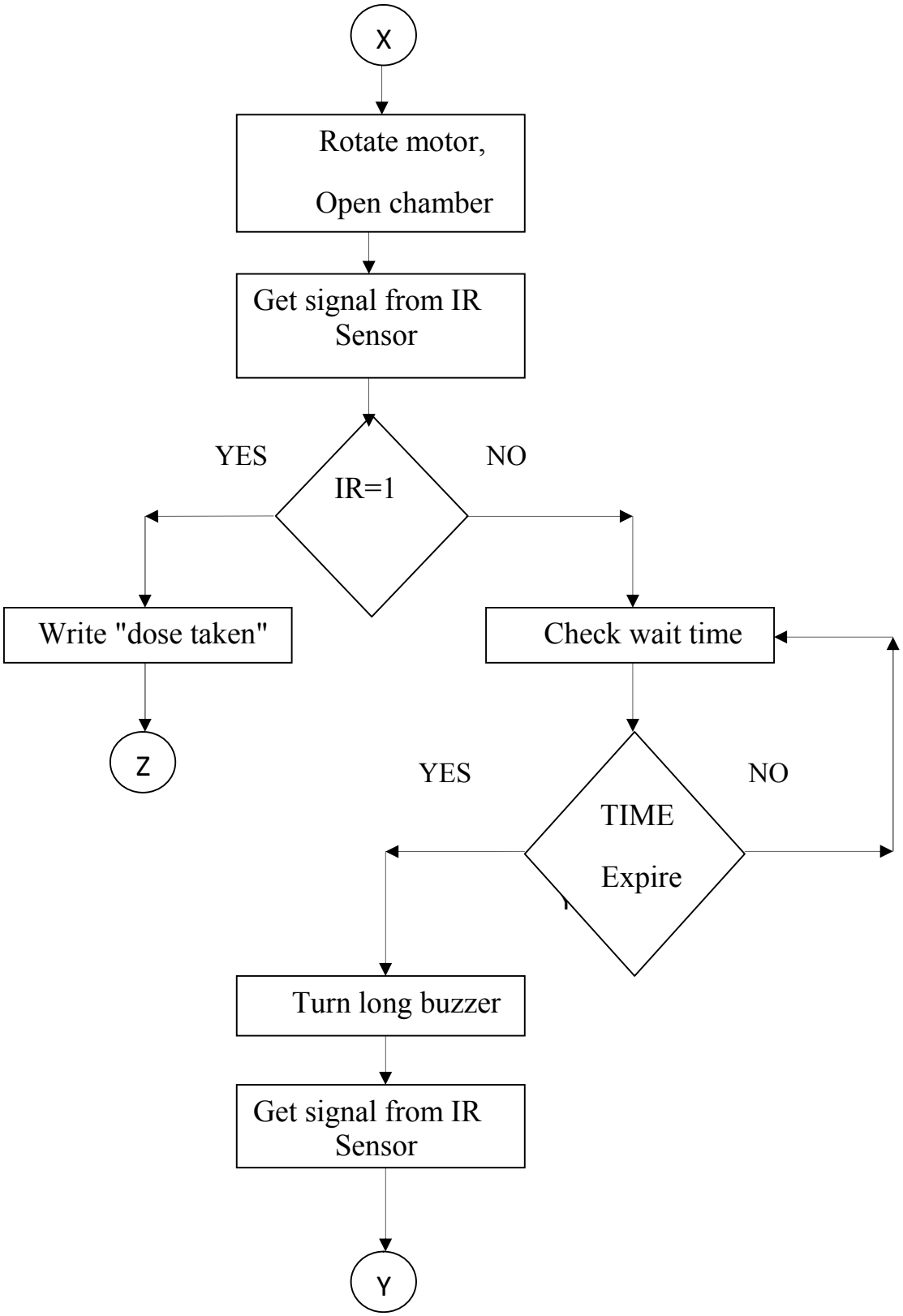
Simulation model under different operation condition has been developed in Proteus plate form. In Proteus simulator it is possible to develop and test such designs before a physical prototype is constructed and user can interact with the design using on screen indicators such as LED and LCD displays and actuators such as switches and buttons. Proteus also provides extensive debugging facilities including breakpoints,

single stepping and variable display for both assembly code and high level language source.

4.1.3 System flow chart

The flow chart illustrates the data flow of the software inside the microcontroller, and describes System operation.




X
Rotate motor,
Open chamber
Get signal from IR Sensor
YES
IR=1
NO
Write "dose taken"
Check wait time
Z
YES
TIME
Expire
NO
Turn long buzzer
Get signal from IR Sensor
Y

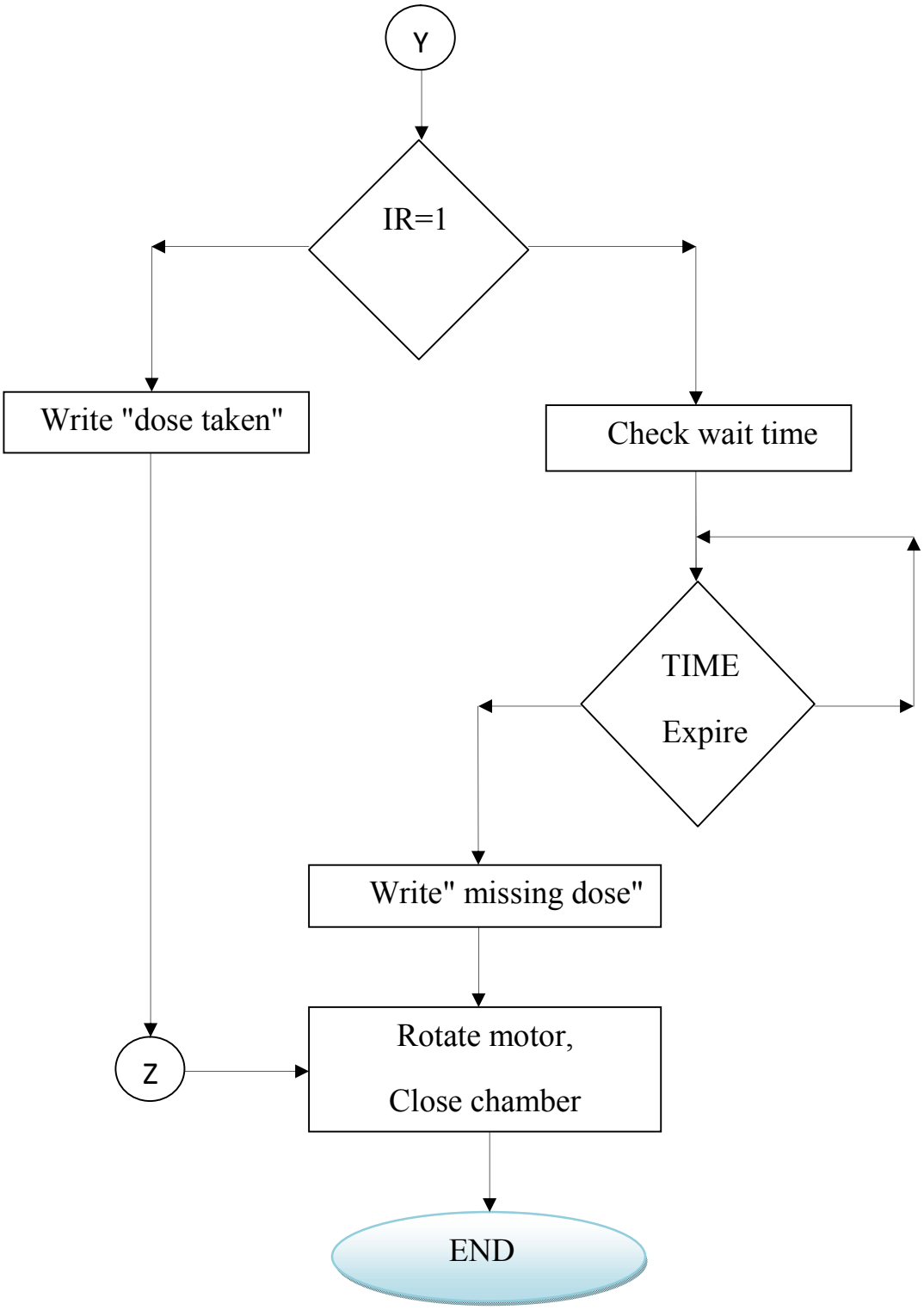


Figure 4.1: system flow chart

4.1.4 The operation of automatic medication dispenser

The operation of automatic medication dispenser starts by promt the user to search the home position where the motor starts the movement. The home position starts by pressing the ON button and stop the system by the same button when reaching the "H" sign. Figure 4.1, shows message in the LCD asking for searching home.

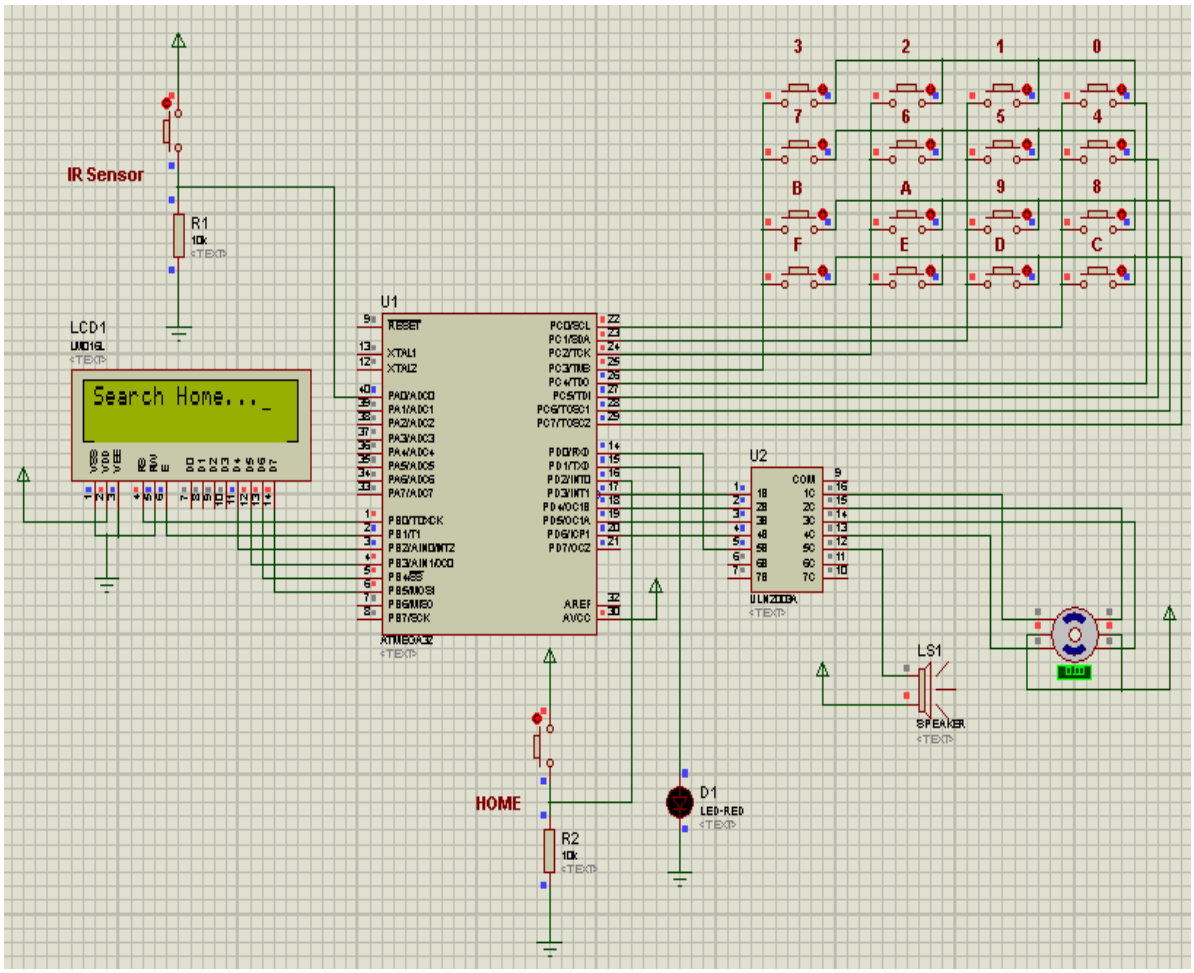


Figure 4.2: Searching home

- **Data input process**

After that the system asks for entering all chambers information which are the tablet time, the start time (the first time entering by considering the user should take his or her medication), number of doses or dose repetition (it is the number of times the medication must be taken per day) and number of tablets in the chamber to know the time of stopping when no more tablets to dispense. Figures (4.3-4.5) show the data

input process. Figure (4.3) shows the step of the tablet time entering. Figure 4.4 shows the step of entering the time between the doses. Figure 4.5 shows the step of entering number of pills in chamber .If any chamber not used Pills have to be set to zero.

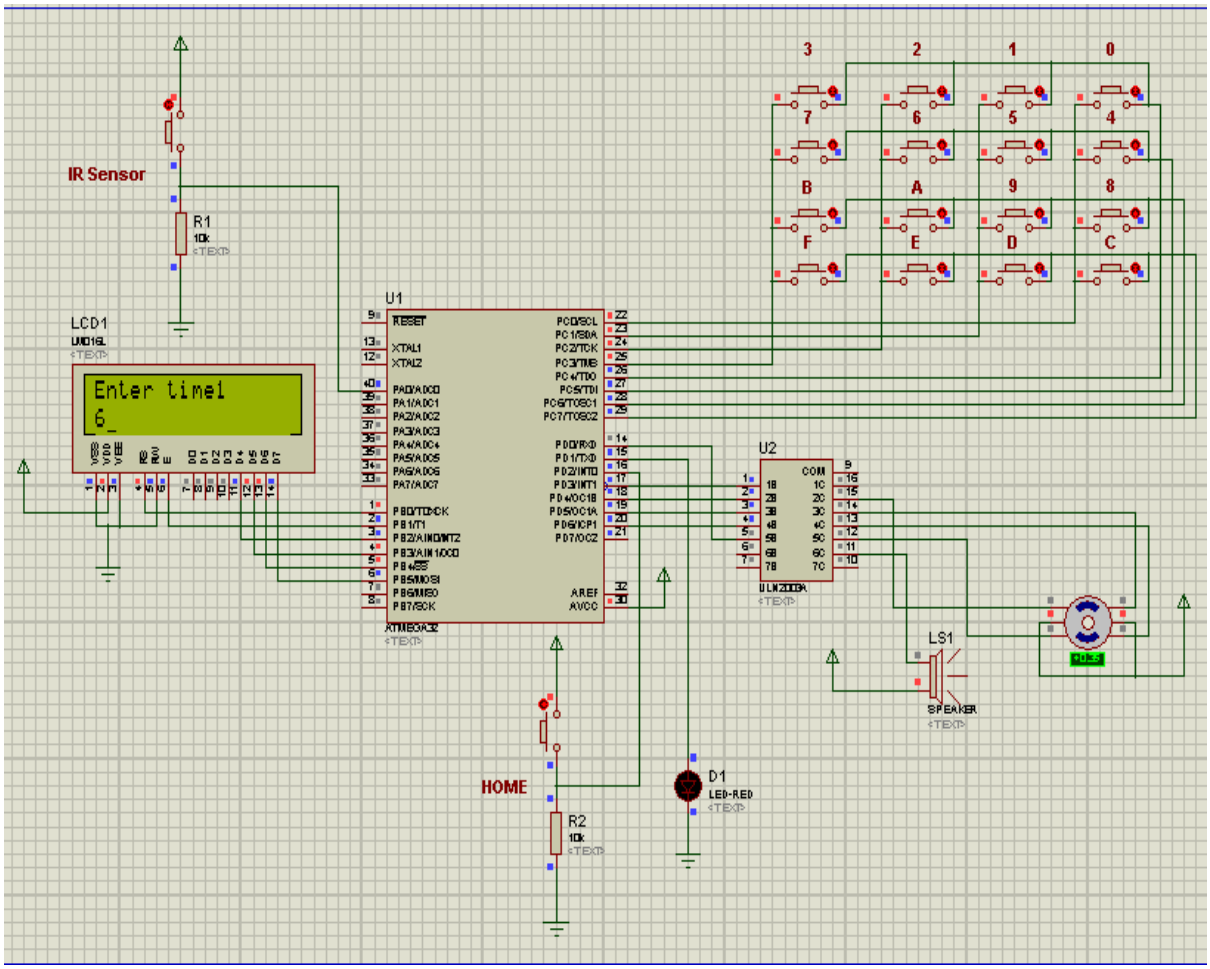


Figure 4.3: Entering dose time

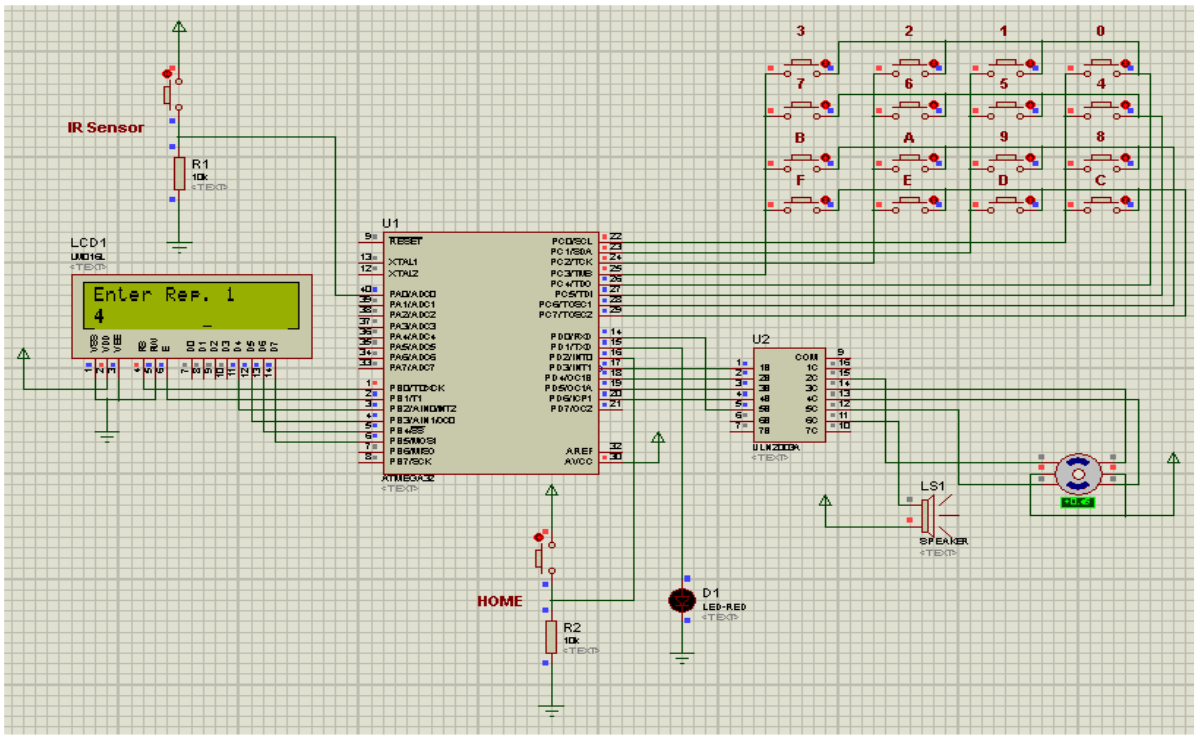


Figure 4.4: Entering tablet repetition

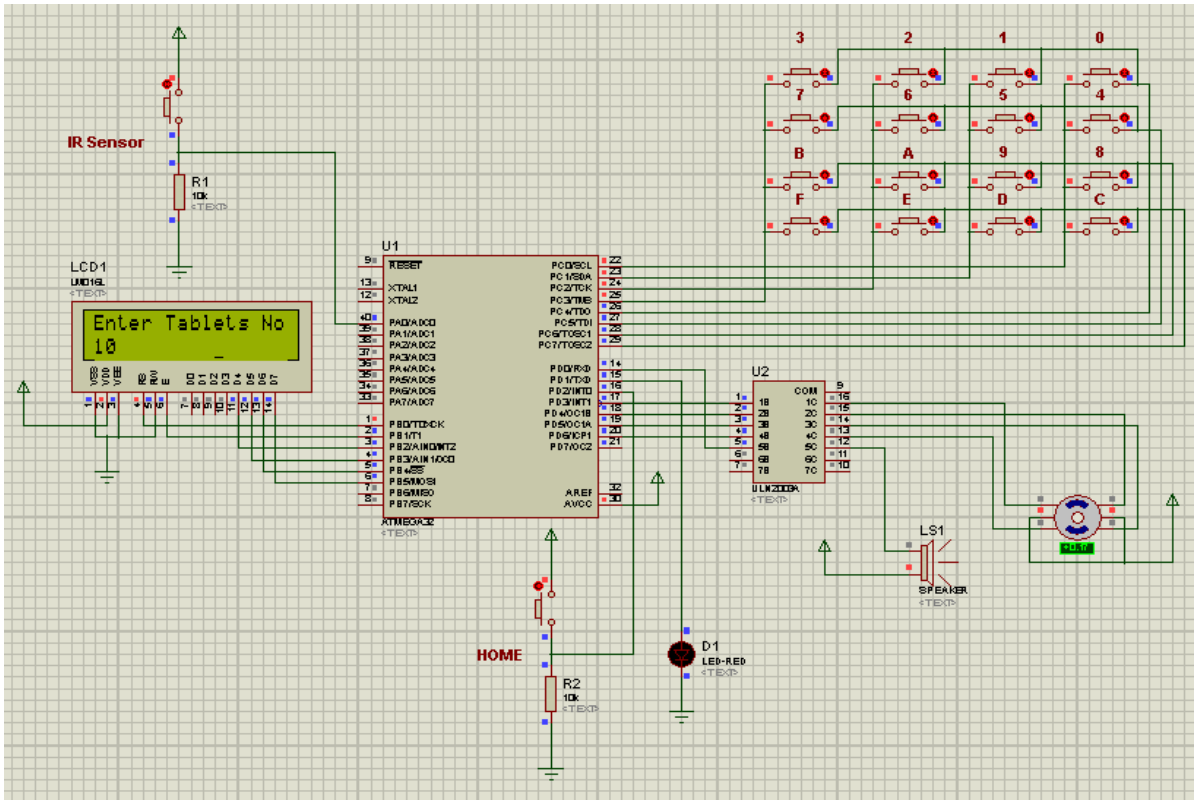


Figure 4.5: Entering number of tablets

Figure 4.6 shows the data input process.

Figure 4.6: Data input process

- **Dispensing process**

The procedure for each chamber is similar to the others, it starts by monitoring the time of each chamber and compare it with a counting up 1-sec base timer as shown in figure 4.7.

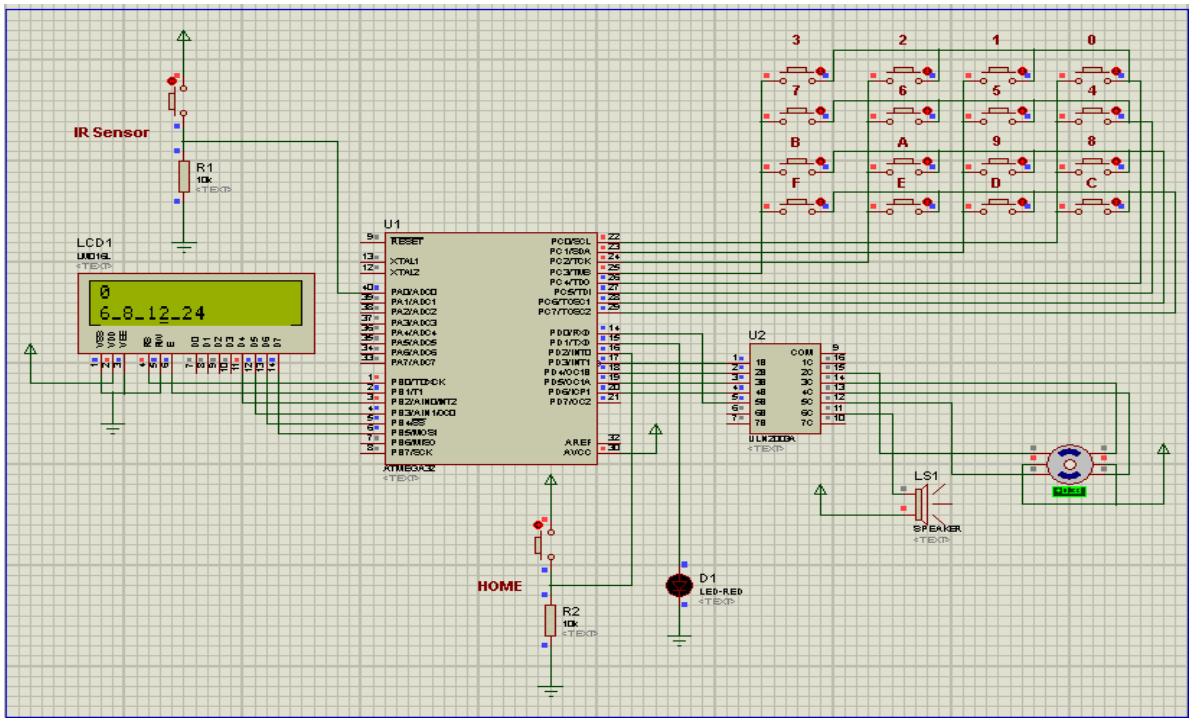


Figure 4.7: 1-Sec base timer counting

When a time for a chamber has elapsed a sound and light alarm generated and the motor rotate specific number steps according to the chamber and stop the gate at it.

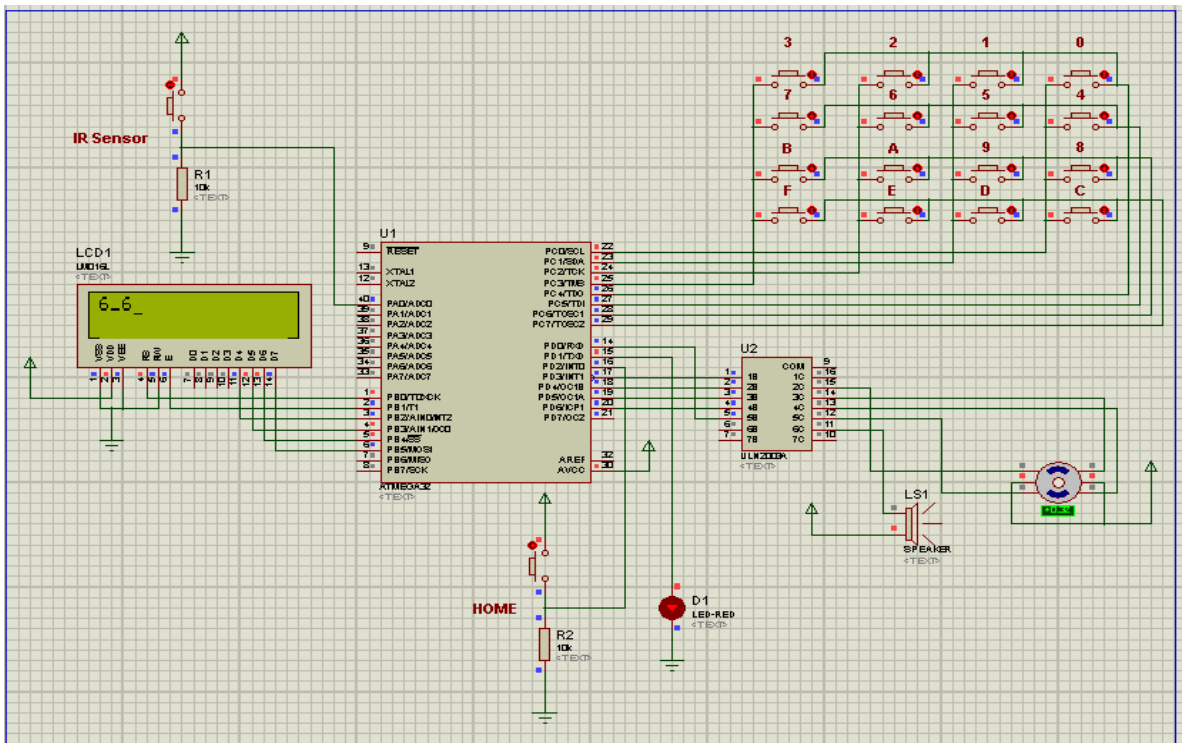


Figure4.6: Dispensing process (a)

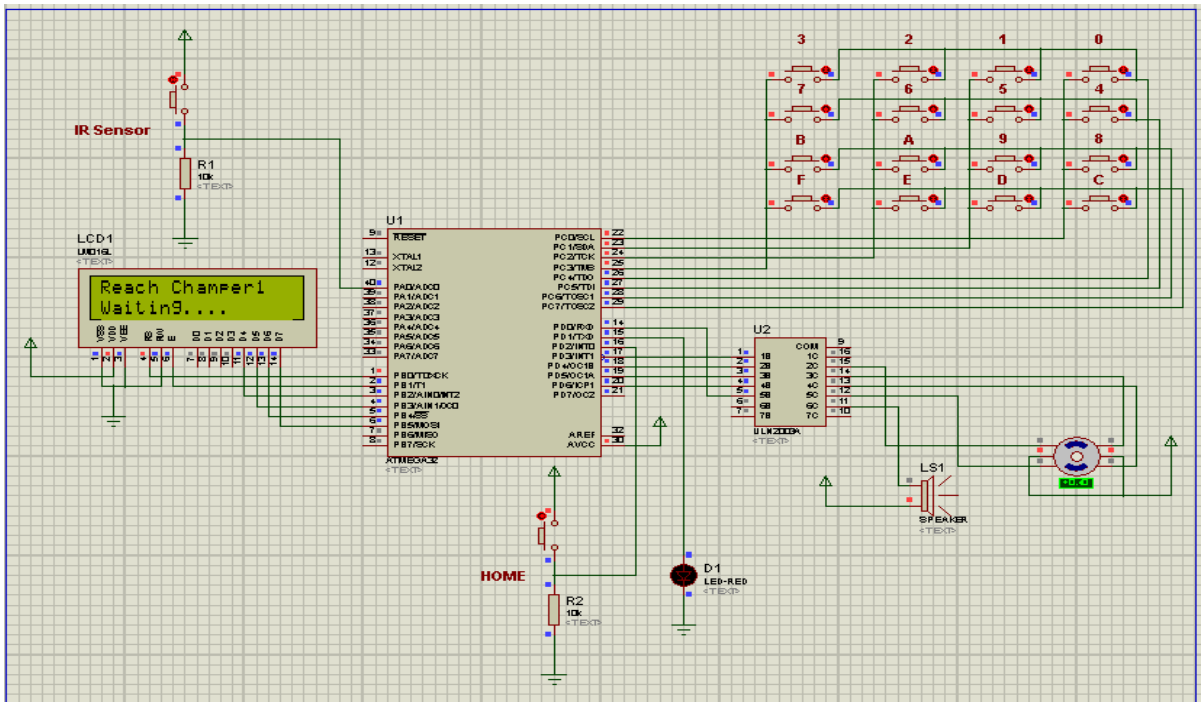


Figure4.7: Dispensing process (b)

The system waits for 10-sec, if the patient pick his dose, the LCD record "dose taken" and the motor wait for a while and return to its home position.

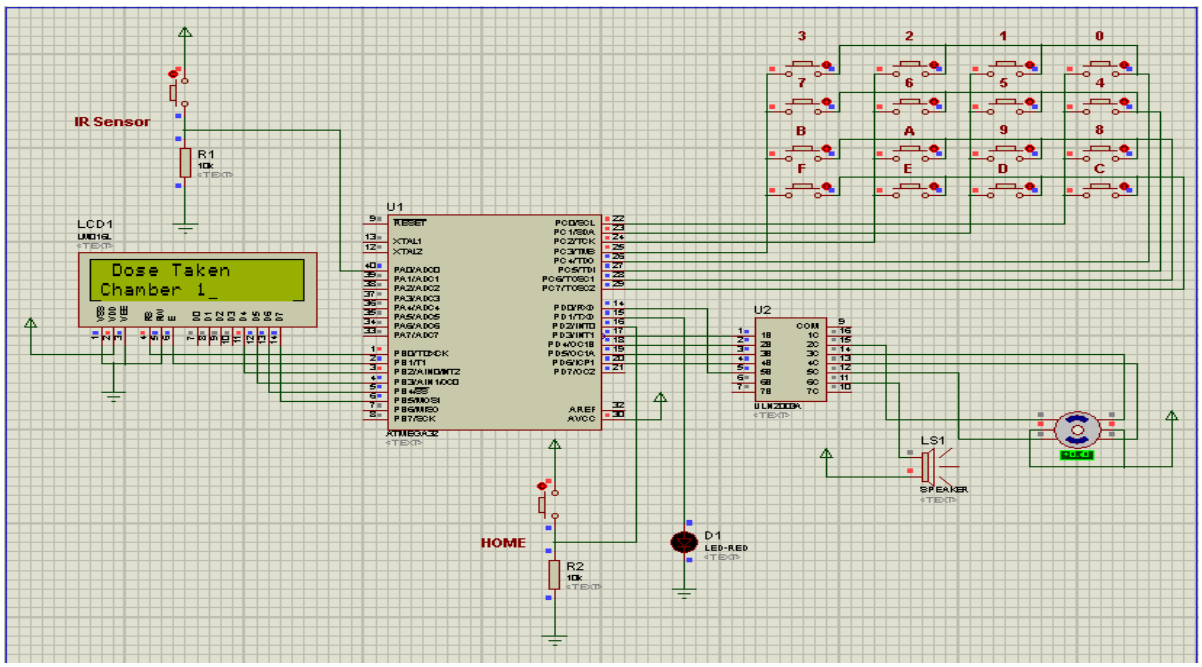


Figure4.8: taken dose case

- **Missing dose process**

If the patient didn't took his dose during this time, a different tone sound alarm generated for longer time. If during this time the dose is taken, the alarm stops and the dose records "dose taken". If no dose is taken, the system record "missing dose", and in all cases the motor returns to home.

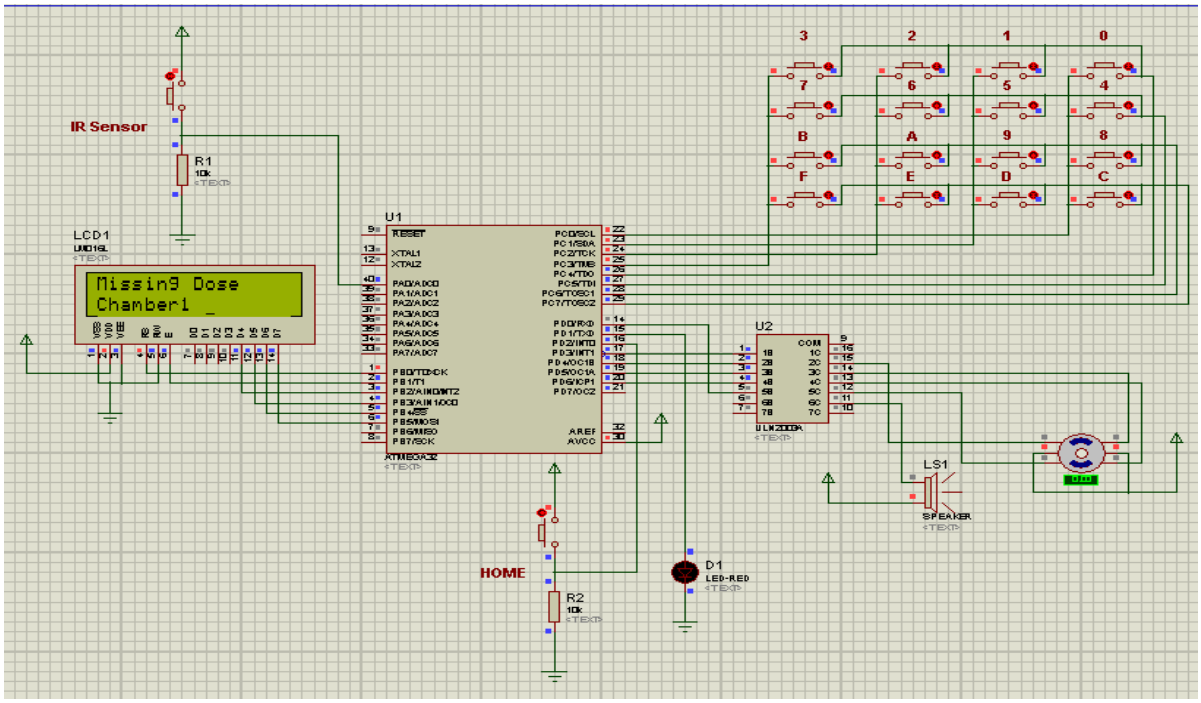


Figure4.9: missing dose case

If only one medication needs to be dispensed the chamber of the corresponding medication will open. If multiple medications need to be dispensed the various medications will be dispensed sequentially after each chamber closed.

4.2 System Hardware

For the system circuit is shown in Figure 4.10.All wires were constantly being tested with digital multimeter and power supply to make sure proper voltages and currents were available. The process of testing, finding problems, attempting to fix the problem, and retesting was repeated over and over until the device functioned properly.

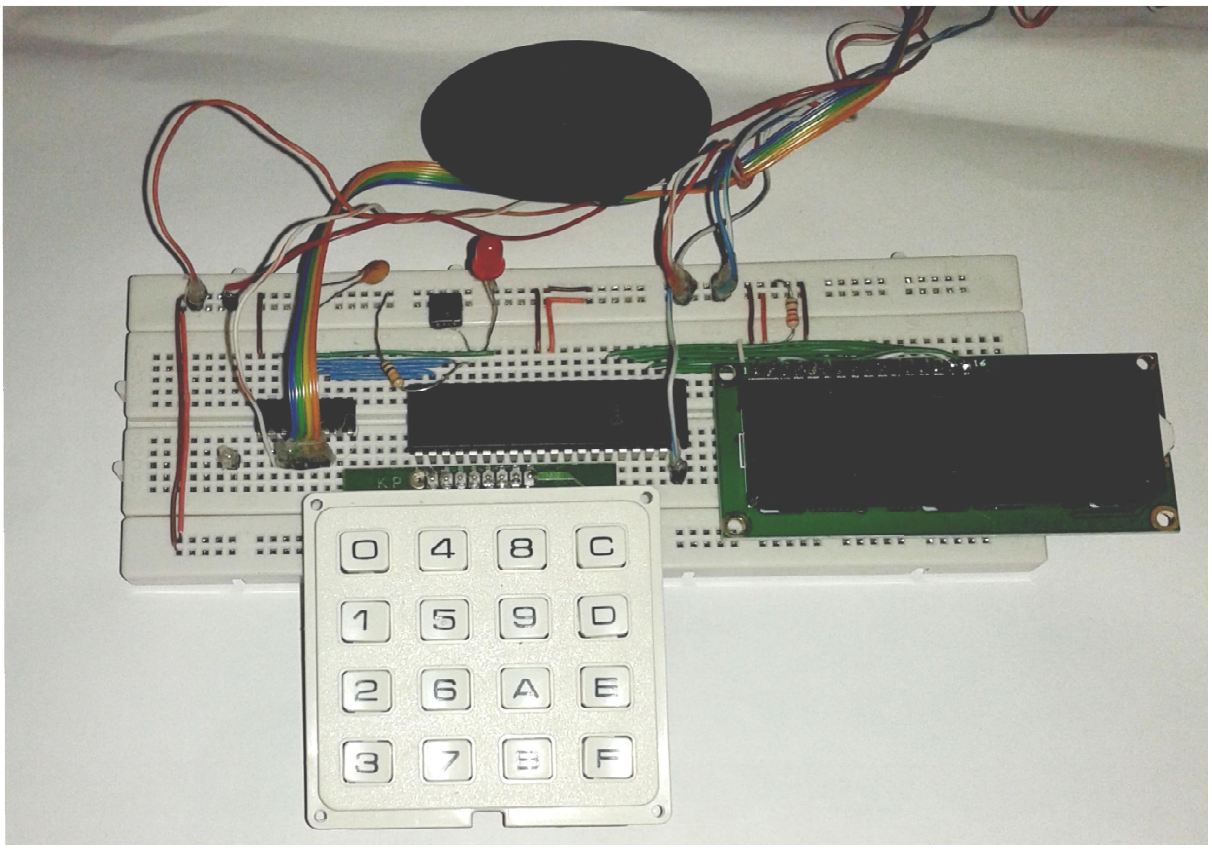


Figure 4.10: Components connection diagram

Chapter Five

Conclusion and Recommendations

5.1 Conclusions

An automatic medication dispenser system has been successfully designed and implemented.

The system is more comfortable as all the medications are stored in the same cabinet and it gives convenience by reducing physical steps and methods so it is helpful for the Elderly population.

The device is a locked system that keeps the medication in a good condition and it serves as safety measure to insure that medication can't be tampered with and it keeps children away as well.

5.2 Recommendation

The following points may be taken as suggested future work:

- The system shall be able to directly communicate with the caregiver in case a noncompliance incident is detected.
- Upgrading the system by adding more container to handle largest number of medication and fit more than one patient
- Interface the microcontroller with network.
- Use memory to save the previous information.

5.3 References

[1] Flynn EA, Barker KN. Medication error research. In: Cohen MR, ed., Medication Errors: Causes and Prevention.

[2] Elizabeth A. Flynn, Ph.D., R.Ph. "A Brief History of Medication Errors" Center for Pharmacy Operations Design, Auburn University, Alabama.

[3] Barker KN, Pearson RE, Helper CD, Smith WE, Pappas KN, "Effect of an automated bedside dispensing machine on medical errors", Am J Hosp Pharm, 1984.

[4]Ben Anderson, Kyle Smith, Matt Strasser, Jason Terhune, "PEZ (Pills EZ) Dispenser", October 2010.

[5]Kevin Villani, Eva Marie Suarez, Jacquelyn Masse rcre," accessible medical device competition", 2005-2006.

[6] Melody Moh, Loc Ho, Zachary Walker, and Teng-Sheng Moh, "A Prototype on RFID and Sensor Networks for Elderly Health Care", in RFID Handbook: Applications, Technology, Security, and Privacy, 2008.

[7]Frederick M.Cady, " Microcontrollers and Microcomputers", Oxford University Press, 1997.

[8] A.P Godse, “Microprocessor and Microcontroller System”, Technical Publication, June 2007.

Appendix A

```
$regfile "m32def.dat"

$crystal = 8000000

Config Lcd = 16 * 2
Config Lcdpin = Pin, Db4 = Portb.2, Db5 = Portb.3, Db6 = Portb.4, Db7 = Portb.5,
E = Portb.1, Rs = Portb.0
Config Adc = Single , Prescaler = Auto , Reference = Avcc
Start Adc
Config Timer1 = Timer , Prescale = 256
Enable Interrupts
Enable Ovf1
On Timer1 1_sec
Config Kbd = Portc , Delay = 200 , Debounce = 100
Config Portd = Output
Config Portd.2 = Input
Cls
Dim X As Byte, I As Integer, J As Integer
Dim W As Word , Y As Single , T As Integer
Dim T1 As Integer , T2 As Integer , T3 As Integer , T4 As Integer , Tm As Integer
Dim Tm1 As Integer , Tm2 As Integer , Tm3 As Integer , Tm4 As Integer , Tms As
Integer , Tmax As Integer
Dim Tc As Integer , Hand As Integer , K As Integer
Dim R1 As Byte , R2 As Byte , R3 As Byte , R4 As Byte , R As Integer
Dim Tb As Byte , Tb1 As Byte , Tb2 As Byte , Tb3 As Byte , Tb4 As Byte
Dim Ch As Byte , Ch1 As Byte , Ch2 As Byte , Ch3 As Byte , Ch4 As Byte
Dim Cntr As Integer , Cntr1 As Byte , Cntr2 As Byte , Cntr3 As Byte , Cntr4 As Byte
Dim Tb_cntr1 As Byte , Tb_cntr2 As Byte , Tb_cntr3 As Byte , Tb_cntr4 As Byte
Dim Steps As Integer , Hand1 As Bit , Home2 As Byte
Stop Timer1
Home2 = 0
Do
```

```
If Home2 = 0 Then Gosub Home1
If Home2 > 1 Then Gosub Times
Loop
'#########################################################
Home1:
Do
Cls
While Pind.2 = 0
X = Getkbd()
Locate 1 , 1
Lcd "Search Home..."
End
While Pind.2 = 1
End
While Pind.2 = 0
Portd.tm = 1
Waitms 5
Portd.tm = 0
Tm = Tm + 1
If Tm > 6 Then Tm = 3
End
Wait 1
Cls
Locate 1 , 1
Lcd "IAM HOME....."
Wait 2
Gosub Times
Loop
'#########################################################
Times:
Home2 = Home2 + 1
Cls
```

```
T = 0
T1 = 0
T2 = 0
T3 = 0
T4 = 0
Hand = 0
Tc = 0
I = 0
Cntr1 = 0
Cntr2 = 0
Cntr3 = 0
Cntr4 = 0
Do
Stop Timer1
For Tms = 1 To 4
Locate 1 , 1
Lcd "Enter time" ; Tms ; "     "
Locate 2 , 1
Gosub Timers
Select Case Tms
Case 1 : Tm1 = Tm
Case 2 : Tm2 = Tm
Case 3 : Tm3 = Tm
Case 4 : Tm4 = Tm
End Select
Locate 1 , 1
Lcd "Enter Rep. " ; Tms
Locate 2 , 1
Gosub Repetition
Locate 1 , 1
Lcd "Enter Tablets No"
Locate 2 , 1
```

```
Gosub Tablets
Next Tms
Gosub Chambers
Loop
'#########################################################
Chambers:
If Cntr1 = R1 And Cntr2 = R2 And Cntr3 = R3 And Cntr4 = R4 Then Gosub Times
Cls
If Tm1 > Tm2 And Tm1 > Tm3 And Tm1 > Tm4 Then Tmax = Tm1
If Tm2 > Tm1 And Tm2 > Tm3 And Tm2 > Tm4 Then Tmax = Tm2
If Tm3 > Tm2 And Tm3 > Tm1 And Tm3 > Tm4 Then Tmax = Tm3
If Tm4 > Tm2 And Tm4 > Tm3 And Tm4 > Tm1 Then Tmax = Tm4
Tmax = Tmax + 2
Start Timer1
Do
If T > Tmax Then T = 0
Locate 1 , 1
Lcd T
Locate 2 , 1
Lcd Tm1 ; "_" ; Tm2 ; "_" ; Tm3 ; "_" ; Tm4
If T = Tm1 And Tm1 > 0 Then
Stop Timer1
Cls
Lcd T ; "_" ; Tm1
'**************************************
Gosub Chamber1234
End If
If T = Tm2 And Tm2 > 0 Then
Stop Timer1
Gosub Chamber1234
End If
If T = Tm3 And Tm3 > 0 Then
Stop Timer1
```

```
Gosub Chamber1234
End If
If T = Tm4 And Tm4 > 0 Then
Stop Timer1
Gosub Chamber1234
End If
Loop
'#############################################################
Chamber1234:
For J = 1 To 10
Portd.1 = Not Portd.1
Sound Portd.0 , 250 , 200
Waitms 200
Portd.0 = 0
Next J
Portd.1 = 0
Cls
Start Timer1
Do
If T > Tmax Then T = 0
If T = Tm1 Then
Stop Timer1
Steps = 550
T = T + 1
Ch = 1
If Tm1 = 0 Then Gosub Chambers
End If
If T = Tm2 Then
T = T + 1
Stop Timer1
Steps = 1815
Ch = 2
```

```
If Tm2 = 0 Then Gosub Chambers
End If
If T = Tm3 Then
T = T + 1
Stop Timer1
Steps = 2960
Ch = 3
If Tm3 = 0 Then Gosub Chambers
End If
If T = Tm4 Then
T = T + 1
Stop Timer1
Steps = 4050
Ch = 4
If Tm4 = 0 Then Gosub Chambers
End If
Wait 1
Lcd "Champer" ; Ch
'*****************************************
Locate 2 , 1
Lcd Cntr ; "_" ; Steps
'*****************************************
K = 3
While Cntr < Steps
Portd.k = 1
Waitms 3
Portd.k = 0
K = K + 1
If K > 6 Then K = 3
'*****************************************
Cntr = Cntr + 1
'Locate 1 , 1
'Lcd Cntr
```

```
'*********************************************
End
Cntr = 0
Waitms 500
Cls
If Ch = 1 And Cntr1 < R1 Then Cntr1 = Cntr1 + 1
If Ch = 2 And Cntr2 < R2 Then Cntr2 = Cntr2 + 1
If Ch = 3 And Cntr3 < R3 Then Cntr3 = Cntr3 + 1
If Ch = 4 And Cntr4 < R4 Then Cntr3 = Cntr3 + 1
Lcd "Reach Champer" ; Ch
Locate 2 , 1
Lcd "Waiting....     "
Wait 1
If Cntr1 = R1 Then Lcd "Cycle Finish CH1"
If Cntr1 < R1 Then Lcd "Pick Tablet     "
If Cntr2 = R2 Then Lcd "Cycle Finish CH2"
If Cntr2 < R2 Then Lcd "Pick Tablet     "
If Cntr3 = R3 Then Lcd "Cycle Finish CH3"
If Cntr3 < R3 Then Lcd "Pick Tablet     "
If Cntr4 = R4 Then Lcd "Cycle Finish CH4"
If Cntr4 < R4 Then Lcd "Pick Tablet     "
'*****************************************************
 Repetitions:

Tc = 0
While Tc < 50
W = Getadc(0)
Y = W * 5
Y = Y / 1024
If Y > 3.6 Then
Hand1 = 1
Hand = Hand + 1
Portd.1 = 1
```

```
End If
If Hand > 0 Then
W = Getadc(0)
Y = W * 5
Y = Y / 1024
While Y > 3.6
W = Getadc(0)
Y = W * 5
Y = Y / 1024
End
Tc = 50
Wait 1
Portd.1 = 0
End If
If Y < 3 Then Portd.1 = 0
Waitms 200
Tc = Tc + 1
End
If Hand > 0 Then
If Ch = 1 Then
Tb_cntr1 = Tb1 - Hand
If Tb_cntr1 < 1 Then
Cls
Lcd "Chamber1 EMPTY"
End If
End If
If Ch = 2 Then
Tb_cntr2 = Tb2 - Hand
If Tb_cntr2 < 1 Then
Cls
Lcd "Chamber2 EMPTY"
End If
```

```
End If
If Ch = 3 Then
Tb_cntr3 = Tb3 - Hand
If Tb_cntr3 < 1 Then
Cls
Lcd "Chamber3 EMPTY"
End If
End If
If Ch = 4 Then
Tb_cntr4 = Tb4 - Hand
If Tb_cntr4 < 1 Then
Cls
Lcd "Chamber4 EMPTY"
End If
End If
End If
'**************************************
If Hand < 1 Then
For I = 1 To 30
Portd.1 = Not Portd.1
Sound Portd.0 , 250 , 300
W = Getadc(0)
Y = W * 5
Y = Y / 1024
If Y < 1 Then Hand1 = 0
If Y > 3.6 Then
I = 30
Hand1 = 1
End If
Waitms 100
Portd.0 = 0
Next I
Portd.1 = 0
End If
Wait 1
```

```
Cls
If Hand1 = 0 Then
Locate 1 , 1
Lcd "Missing Dose"
Locate 2 , 1
Lcd "Chamber " ; Ch
Wait 1
End If
If Hand1 = 1 Then
Locate 1 , 1
Lcd " Dose Taken"
Locate 2 , 1
Lcd "Chamber " ; Ch
Wait 1
End If
Hand = 0
K = 6
For I = 0 To Steps
Portd.k = 1
Waitms 3
Portd.k = 0
K = K - 1
If K < 3 Then K = 6
Next I
Start Timer1
Gosub Chambers
'*************************************
Loop
1_sec:
T = T + 1
Tc = Tc + 1
Timer1 = 31250
```

```
Return
'QQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQ
Timers:
T = 0
T1 = 0
T2 = 0
T3 = 0
T4 = 0
Hand = 0
Tc = 0
I = 0
Do
X = Getkbd()
While X <> 15
X = Getkbd()
While X = 16
X = Getkbd()
End
If X <> 15 Then
I = I + 1
Select Case I
Case 1 : T1 = X
Case 2 : T2 = X
Case 3 : T3 = X
Case 4 : T4 = X
End Select
Lcd X
End If
End
Cls
'*****************************************
If I = 4 Then
T1 = T1 * 1000
```

```
T2 = T2 * 100
T3 = T3 * 10
End If
If I = 3 Then
T1 = T1 * 100
T2 = T2 * 10
End If
If I = 2 Then
T1 = T1 * 10
End If
Tm = T1 + T2
Tm = Tm + T3
Tm = Tm + T4
I = 0
Return
Loop
'#########################################################
Repetition:
Do
X = Getkbd()
While X <> 15
X = Getkbd()
 While X = 16
X = Getkbd()
End
If X <> 15 Then
R = X
Lcd R ; "        "
End If
Wend
Select Case Tms
Case 1 : R1 = R
```

```
Case 2 : R2 = R
Case 3 : R3 = R
Case 4 : R4 = R
End Select
Return
Loop
'#########################################################
Tablets:
Do
X = Getkbd()
While X <> 15
X = Getkbd()
 While X = 16
X = Getkbd()
End
If X <> 15 Then
Tb = X
Lcd Tb; "        "
End If
End
Select Case Tms
Case 1: Tb1 = Tb
Case 2: Tb2 = Tb
Case 3: Tb3 = Tb
Case 4: Tb4 = Tb
End Select
Return
Loop
```

Appendix B

B.1 Microcontroller pin connection

Pin Number	Pin name	Function	Connected to
1	PB0	OUTPUT	LCD(RS)
2	PB1	OUTPUT	LCD(E)
3	PB2	OUTPUT	LCD(D4)
4	PB3	OUTPUT	LCD(D5)
5	PB4	OUTPUT	LCD(D6)
6	PB5	OUTPUT	LCD(D7)
7	PB6		
8	PB7		
9	RESET		
10	VCC	VCC	+5 V
11	GND	GND	0 V
12	XTAL2		

13	XTAL1		
14	PD0	OUTPUT	ULN2003(B5)
15	PD1	OUTPUT	LED-RED
16	PD2	INPUT	
17	PD3	OUTPUT	ULN2003(B1)
18	PD4	OUTPUT	ULN2003(B2)
19	PD5	OUTPUT	ULN2003(B3)
20	PD6	OUTPUT	ULN2003(B4)
21	PD7		
22	PC0	INPUT	KEYPAD
23	PC1	INPUT	KEYPAD
24	PC2	INPUT	KEYPAD
25	PC3	INPUT	KEYPAD
28	PC6	INPUT	KEYPAD
29	PC7	INPUT	KEYPAD

30	AVCC	VCC	+5 V
31	GND	GND	0 V
32	AFREE		
33	PA0	INPUT	IR RECEVIER LED
35	PA2		
36	PA3		
37	PA4		
38	PA5		
39	PA6		
40	PA7		

B.2 Pin description of LCD

Pin Number	Function	Connected to
1	VSS	
2	VDD	
3	VEE	
4	RS	PB.0
5	RW	GROUND
6	E	Pin B.1
7	D0	
8	D1	
9	D2	
10	D3	
11	D4	PB2
12	D5	PB3
13	D6	PB4
14	D7	PB5

B.3 ULNA2003 Connection

Pin number	Pin name	Function	Connected to
1	B1	INPUT	ATMEGA32(PD3)
2	B2	INPUT	ATMEGA32(PD4)
3	B3	INPUT	ATMEGA32(PD5)
4	B4	INPUT	ATMEGA32(PD6)
5	B5	INPUT	ATMEGA32(PD0)
6	B6		
7	B7		
8	E		
9	COM		
10	C1		
11	C2	OUTPUT	STEPPER MOTOR
12	C3	OUTPUT	STEPPER MOTOR
13	C4	OUTPUT	STEPPER MOTOR

14	C5	OUTPUT	STEPPER MOTOR
15	C6	OUTPUT	SPEACKER
16	C7		