INFORMATION TECHNOLOGY

Household Vision Tester

Group No: 01 - Batch-17

	Index No	Name
Leader	**174015X**	**M.A.M.Bandara**
Member 1	174149K	G.C.Sarujan
Member 2	174154V	T.H.K.Sewwandi
Member 3	174172A	S.Thayana
Member 4	174008E	G.S.Alahakoon

Supervisor Name : Mr.D.K. Withanage Date of submission: 13/01/2020

Mr. B.H. Sudantha

Supervisor signature:

Table of Contents

List of Figures

1.0 Introduction

The house hold vision tester is a system that enables checking of the help of highly stabilized and rapidly functionalized electronic and mechanical control system. Current 253 million people live with vision impairment worldwide among them 36 million are blind and 217 million have moderate or severe vision impairment. Some of the common eye disorders are Hyperopia (long sightedness), Myopia(shortsightedness) and color blindness. Colorblindness is a genetic disorder. In individuals with Northern European ancestry, as many as 8% of men and 0.4% of women experience congenital color deficiency.

So in our system can identify the color blindness and long vision in their home. Therefor they will reduce their time to go hospital and look after their health. Our system will use full to check eye without other person help.

2.0 Literature Survey

There are some online vision checks which are used to check the visual acuity check, contrast vision check and color vision check. And also some researches have been done to identify some eye diseases automatically.

Eg :- automatic diabetic retinopathy detection in smartphone-based fundus photography using artificial intelligence, retinal imaging using machine learning techniques for computer vision.

3.0 Aim and Objectives

3.1 Aim

To develop an automated house hold vision tester system for checking long sightedness, and color blindness at home.

3.2 Objectives

- To check the distant vision by using LCD display and keypad.
- To check the color blindness by using LCD display and keypad.
- To change the distant vision board to color blindness board using servo motor.
- To set the distance between eye and display(board) using gear motor and limit switch.
- Display the test results by using LCD display.

4.0 Analysis and Design

4.1 System Block Diagram

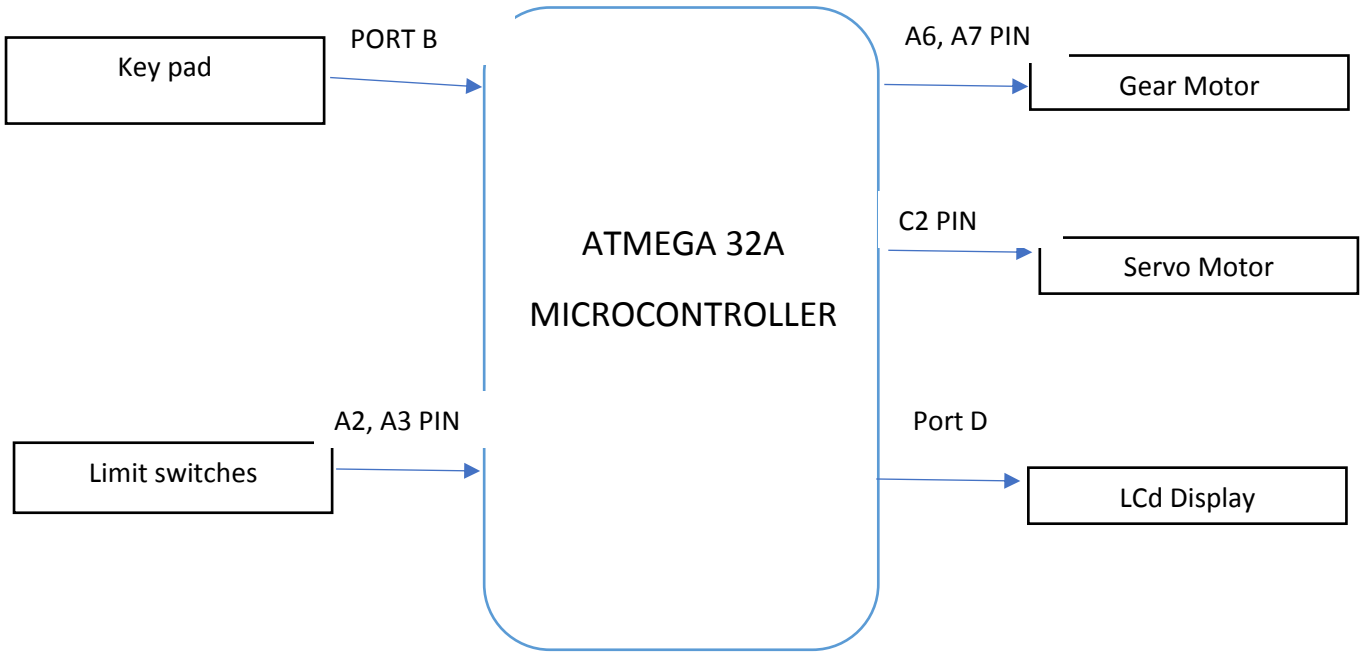


Figure 1 System Diagram

4.2 3D Diagram

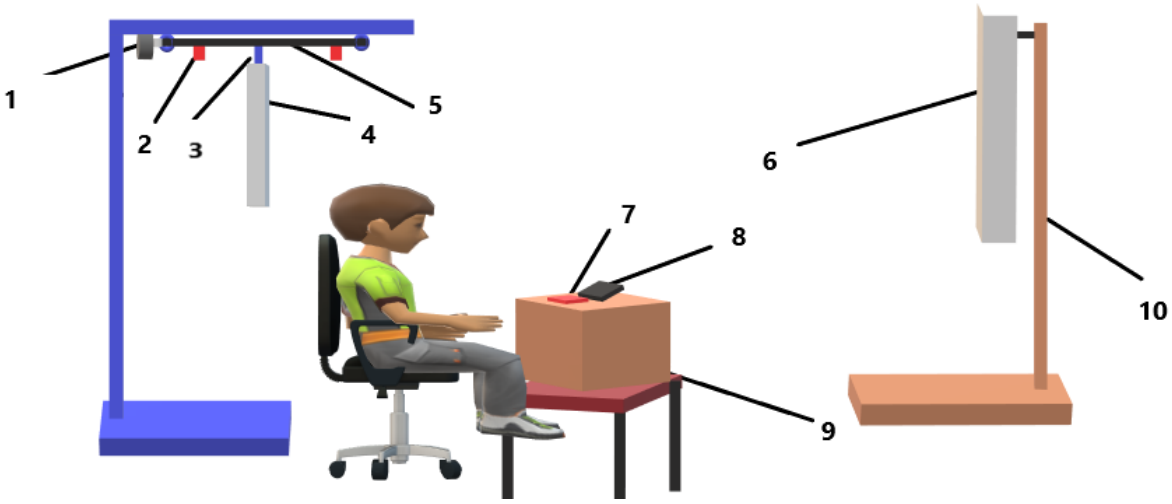


Figure 2 3D diagram

1 - Gear motor
2 - Limit Switch
3 - Servo motor
4 - Vision board
5 - Moving rod
6 - Mirror
7 - Key pad
8 - LCD display
9 - Circuit box
10- Mirror stand

When the machine is switched on, instructions for the patient will be displayed on the LCD display attached to the keypad. First the distant vision test is carried out. Here a distance of 3 m is maintained between the patient and the mirror. Here each number in the chart will be highlighted and the patient should press the number identified in the keypad. For each letter 1 minute is given. After that the patient is instructed to turn back. After that board moves backwards by 25 cm from the patient. Then the color blindness test is carried out by highlighting each number. Here the patient is given 1 minutes for each numbers to input the numbers to the keypad. Finally, the results of the tests are displayed in the LCD display

5.0 Testing and Implementation

First the programming of microcontrollers using Atmel Studio 7.0 was done by us. Then the study of how to stimulate the simple basic circuits using Proteus 8.0. Inorder to test the long vision test and the colour blindness, the vision chart board which is having both the long vision and color blindness charts on opposite sides of the board was prepared.

Figure 3 Long Vision Board

Figure 4 Colour BlindnessBoard

5.1 Automatically changing the long vision board to color blindness board

The mechanical part of automatically changing the distant vision board to color blindness board was done by using servo motor. First the circuit for servo motor was designed using Atmega32. Then the code for the servo motor was written in Atmel studio. Then written program was tested by connecting the vision board to the servo motor.

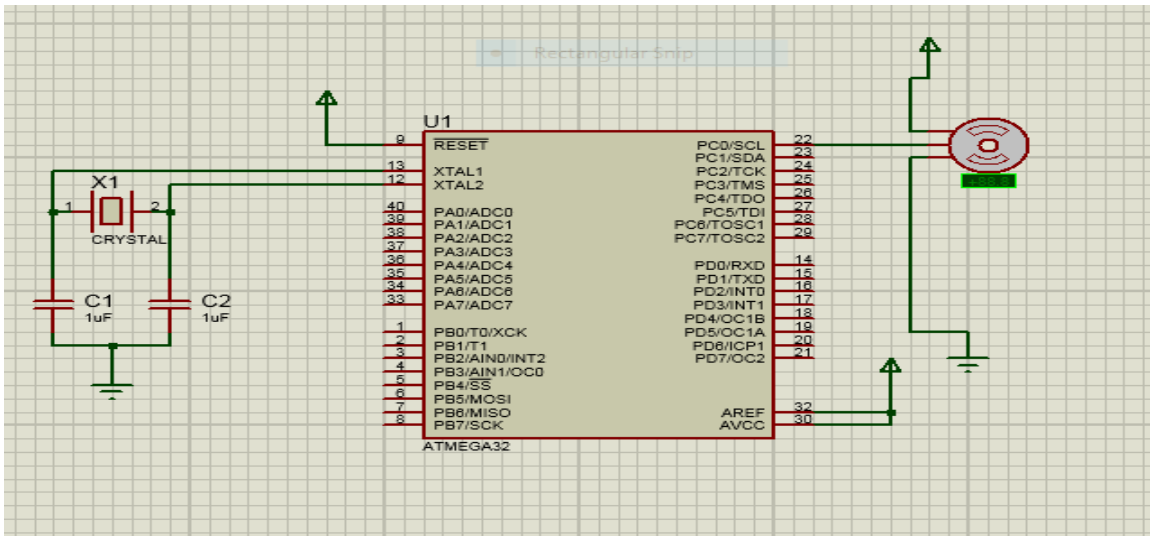


Figure 5 Proteus diagram of Servo Motor

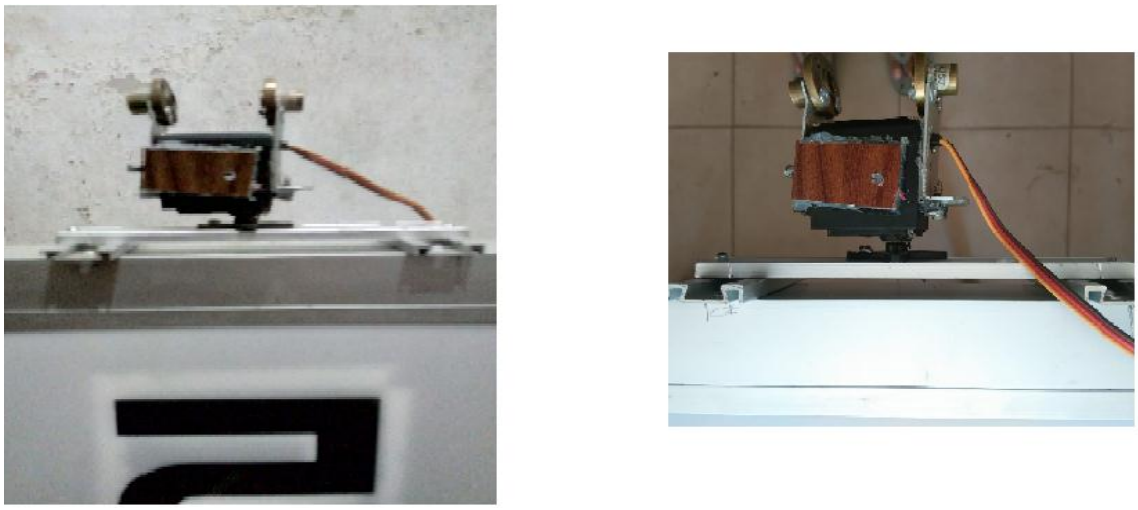


Figure 6 testing servo Motor

Code for servo Motor to Change the distant vision board to color blindness board (In our final circuit atmega c2 pin was used for servo motor)

```
        void board_rotate_front(){
        for(int i=0;i<12;i++){
                PORTC |= (1<<PC2);
                _delay_us(1700);
                PORTC &= ~(1<<PC2);
                _delay_us(18300);
        }
}
        void board_rotate_back(){
        for(int i=0;i<12;i++){
                PORTC |= (1<<PC2);
                _delay_us(900);
                PORTC &= ~(1<<PC2);
                _delay_us(19100);
        }
}
```

5.2 Adjusting the distance between eye and display(board) using gear motor and limit switch

The distance between eye and display(board) before the colour blindness test and long vision test was established using gear motor and limit switch. First the circuit for gear motor with limit switch was designed. Then the program for the gear motor with limit switch was written in Atmel studio. Gear motors need more power and as it is unable to control those motors using microcontroller, a L298N motor driver was used. Here a power pack which can give 12v was used to supply power to motor driver.

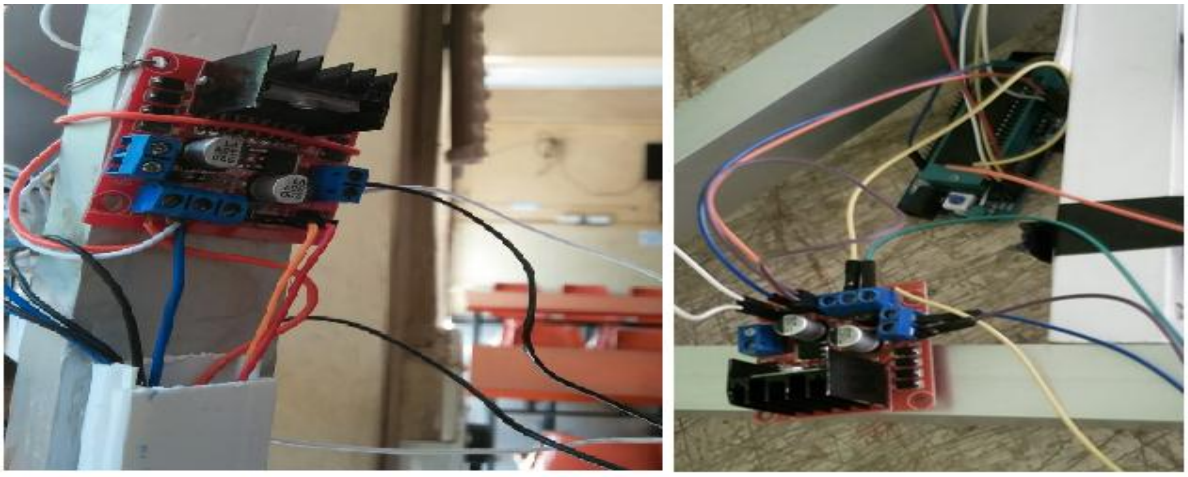


Figure 7 L298N Motor Driver

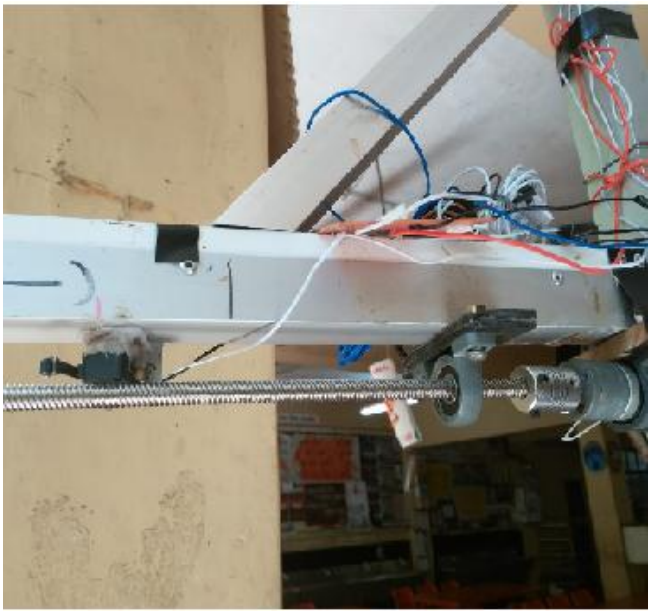
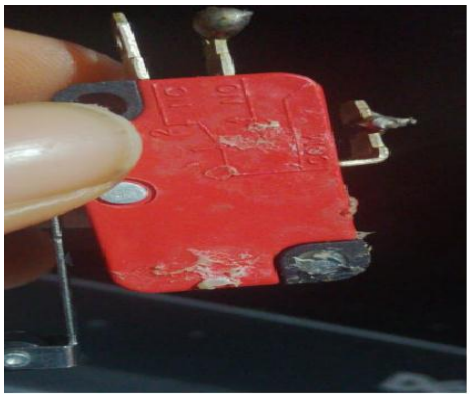


Figure 8 Testing gear motor



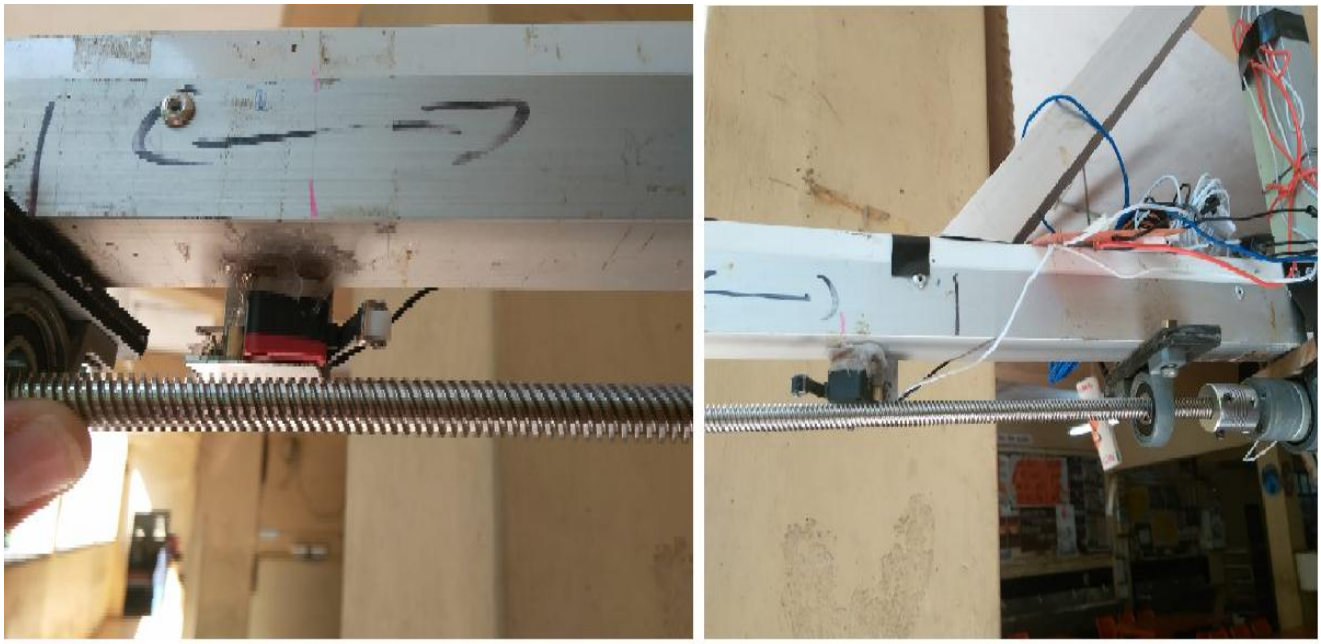


Figure 9 Testing limit switch

Code for gear motor and limit switch

```
////Gear motor
void board_set_front(){
        PORTD |= (1<<PD2);
        PORTD &= ~(1<<PD3);
        //while(!(PIND & (1<<PD0)));
        _delay_ms(5000);
        PORTD &= ~(1<<PD2);
}
//Limit switch
void backward(){
        PORTA &= ~(1<<GEAR_A);
        PORTA |= (1<<GEAR_B);
        while(PINA & 0x04){
                board_rotate_back();
        }

        PORTA &= ~((1<<GEAR_B) | (1<<GEAR_A));
}
```

5.3 Display the test results by using LCD display

Inorder to give the instructions of the band to display the results to the patient a 20*4 LCD display was used. And at the same time a keypad was used to provide inputs.

Figure 10 Testing LCD display and keypad

5.4 Test the vision board

For both vision test charts LED bulbs were fixed behind each letter in the chart and they were connected to the microcontroller via shift registers.

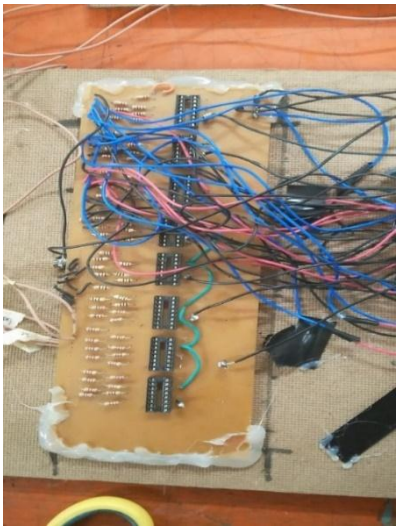


Figure 11 PCB of shift register

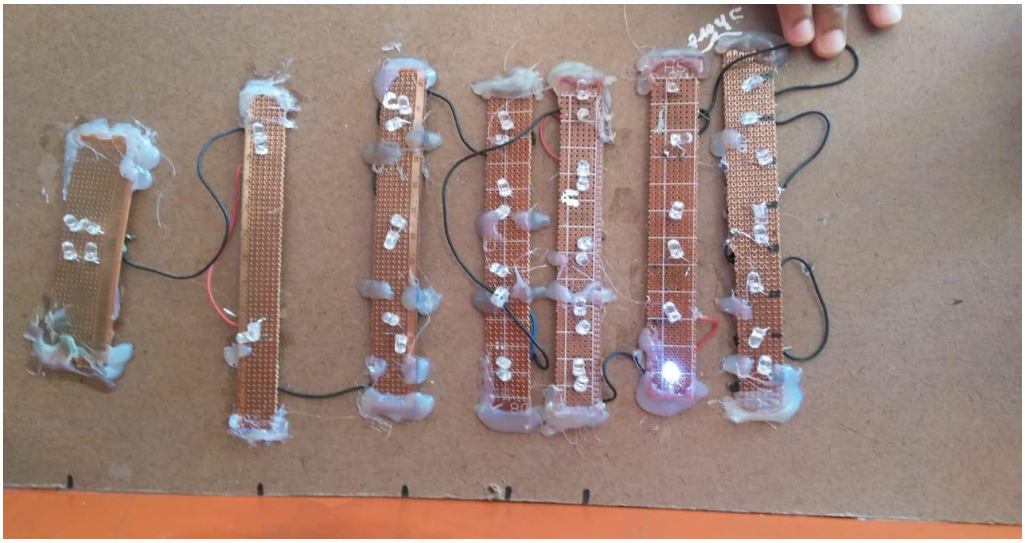


Figure 12 Testing LED using shift register

5.5 Final Source code and circuit

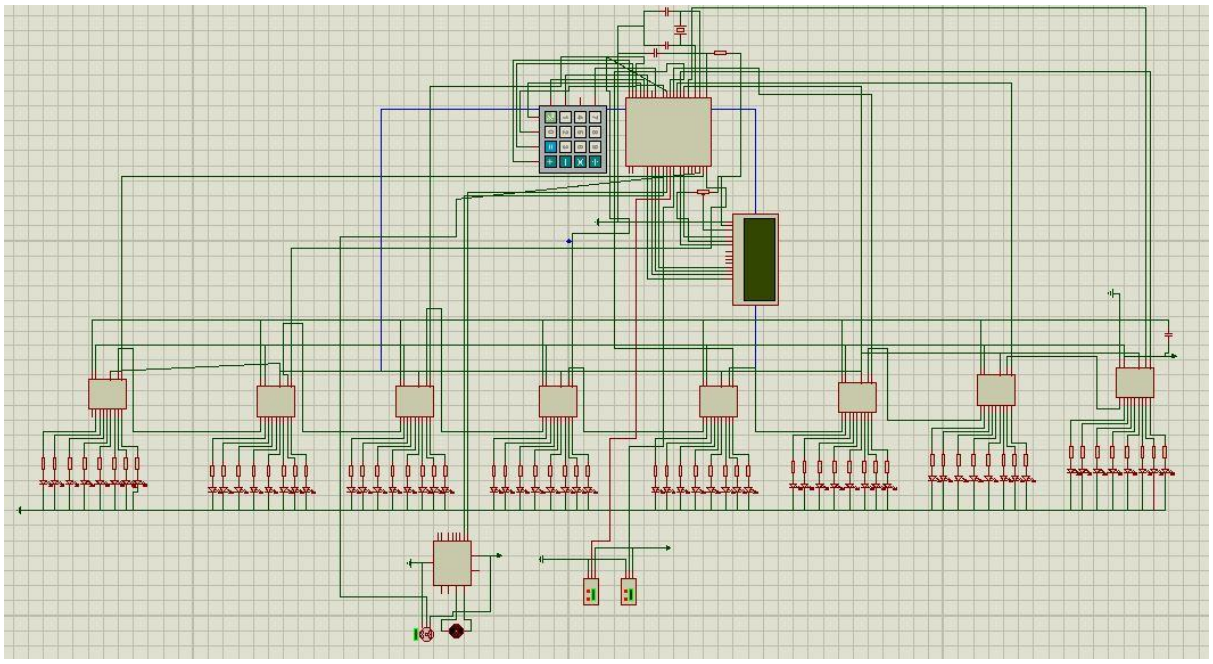


Figure 13 final circuit designed using proteous software

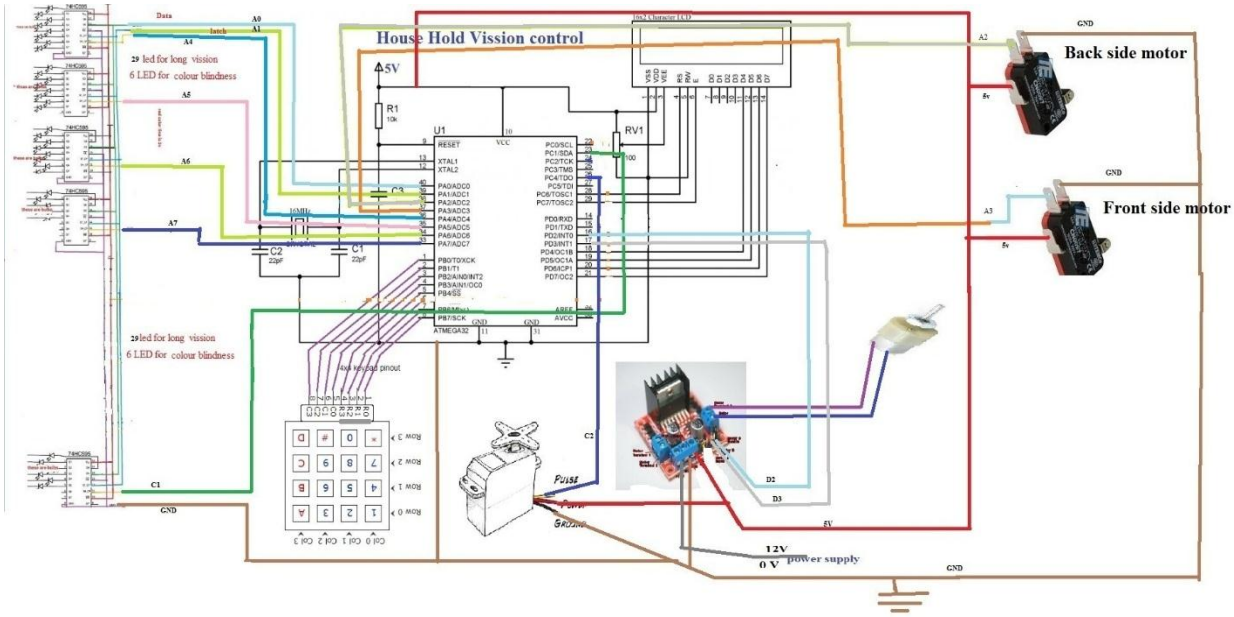


Figure 14 final circuit diagram

Source code

```
/*

final Code

keypad port B

moter out  PA6, PA6

lcd  PD4 – PD7, PC6, PC7

vision bord bulb  port A

Limit switch

        data-> a0

        latch -> a1

        shift 1

        shift 2

        shift 3

        shift 4

        shift 5

        shift 6

        shift 7

        shift 8


servo C2

Limit switch PA3, PA3

*/


# define F_CPU 8000000UL

#define D4 eS_PORTD3

#define D5 eS_PORTD5

#define D6 eS_PORTD6

#define D7 eS_PORTD7

#define RS eS_PORTC6

#define EN eS_PORTC7


#include <avr/io.h>

#include <util/delay.h>
```

```
#include <inttypes.h>
#include <avr/interrupt.h>
#include <stdio.h>
#include <stdlib.h>
#include "lcd.h"
#include <string.h>
#include <math.h>
#include "keypad.h"
#include "shift.h"

#define LIMIT_FRONT PA3
#define LIMIT_BACK PA2
#define GEAR_A PA6
#define GEAR_B PA7

void forward();
void backward();
void initPorts();
int limitSwitch(int);
void gear(int a, int b);

void servo();

char str[4], op;
int count =0;
static volatile int pulse = 0;
static volatile int i = 0;

void shift_init();
void shift_latch();
void shift_high_bit();
void shift_low_bit();
void shift_pulse(uint8_t);
void shift_write(uint8_t,uint8_t);
```

```
void board_set_front();
void board_set_back();
void board_rotate_front();
void board_rotate_back();

void convertstr(int a){

	itoa(a,str,10);
}

char getkey(){
	char a='N';
	while(a=='N'){
		a=scan_for_key();
	}
	return a;
}



char vision1[29] = {
'5',
'9','4',
'2','6','0',
'3','8','9','5',
'0','6','2','4','8',
'6','3','0','9','5','2'
};


void print(char *arr){
	Lcd4_Clear();
	Lcd4_Write_String(arr);
}
```

```
void shift_reset(){
        HC595Write(0x00);
        HC595Write(0x00);
        HC595Write(0x00);
        HC595Write(0x00);
}

void shift_new(uint32_t x){
        HC595Write((x >> 24) & 0xFF);
        HC595Write((x >> 16) & 0xFF);
        HC595Write((x >> 8) & 0xFF);
        HC595Write((x) & 0xFF);
}

char keyRead(){
        char x = 'l';
        while(x=='l' || x=='N'){
                x = scan_for_key();
        }
        return x;
}

char* vision_score[8]={"0 - Fail","6/60","6/36","6/24","6/18","6/12","6/9","6/6"};

// Eye Test
void eyeTest(){
        shift_reset();
        char arr[10];
        char v;

        int lim = 1;
        int count = 0;
```

```
int wrong = 0;

uint32_t x = 2;
for(int i=0;i<21;i++){

        count++;
        if(i==16){
                x*=2;
                i++;
        }
        if(i==7){
                i++;
        }
        shift_new(x);
        print("Enter value");
        v = keyRead();

        if(v == vision1[i]){
                print("Correct");
        }else{
                print("Incorrect");
                wrong++;
        }

        if(wrong*2 >= lim){
                print("Test Done");
                _delay_ms(4000);
                sprintf(arr,"Score=%s",vision_score[lim-1]);
                print(arr);
                _delay_ms(4000);
                break;
        }

        if(count+1 > lim){
```

```
                    lim += 1;
                    count = 0;
                    wrong = 0;
                }
                //sprintf(arr, "char=%c", vision1[i]);
                //print(arr);
                x *= 2;
                _delay_ms(1000);
            }
}
// Lcd display
void showresults(){

        Lcd4_Clear();
        Lcd4_Write_String("your Result :- ");
        convertstr(count);
        Lcd4_Write_String(str);
        _delay_ms(3000);
        Lcd4_Clear();
        count=0;



}

void colorTest(){
        print("Color test");
        _delay_ms(2000);
        shift_reset();
        uint32_t x = pow(2,25);
        char c;
        char arr[10];
        for(int i=0;i<6;i++){
            if(i==5){
                x *= 2;
```

```
                }

                // see - 1 not see - 0

                shift_new(x);
                print("Enter value");
                c = keyRead();

                if(i==0 || i==4){
                        if(c == '0'){
                                print("Correct");
                                _delay_ms(2000);
                        }else{
                                print("Incorrect");
                                _delay_ms(2000);
                        }
                }else{
                        if(c == '1'){
                                print("Correct");
                                _delay_ms(2000);
                        }else{
                                print("Incorrect");
                                _delay_ms(2000);
                        }
                }

                //print(arr);
                x *= 2;
                _delay_ms(2000);
        }
}


int main(){
```

```
DDRD = 0xFF;
DDRC = 0xFF;
DDRB = 0xf0;  // for keybord
DDRA = 0xFF;// for bulb
DDRA &= ~(0<<LIMIT_BACK); //Limit switch
DDRA &= ~(0<<LIMIT_FRONT);  ////Limit switch
PORTD = 0x00;
PORTA = 0x00;
PORTA |= (1<<LIMIT_FRONT) | (1<<LIMIT_BACK);
PORTC = 0x00;

GICR|=(1<<INT0);
MCUCR|=(1<<ISC00);

TCCR1A = 0;

Lcd4_Init();
_delay_ms(1000);

shift_reset();

Lcd4_Clear();
Lcd4_Write_String("hello");
_delay_ms(2000);

forward();

char buff[10];
uint32_t m = 1;
while(1){
        eyeTest();
        backward();
        _delay_ms(2000);
        colorTest();
```

```
                _delay_ms(2000);
                forward();
        }
}


//Limit swith
void forward(){
        PORTA |= (1<<GEAR_A);
        PORTA &= ~(1<<GEAR_B);
        while(PINA & 0x08){
                board_rotate_front();
        }

        PORTA &= ~((1<<GEAR_B) | (1<<GEAR_A));
}
////Limit swith
void backward(){
        PORTA &= ~(1<<GEAR_A);
        PORTA |= (1<<GEAR_B);
        while(PINA & 0x04){
                board_rotate_back();
        }

        PORTA &= ~((1<<GEAR_B) | (1<<GEAR_A));
}

void shift_init(){
        DDRA = 0xFF;
        DDRC |= 0x03;
        PORTA = 0x00;
}

void shift_latch(){
```

```
        PORTA |= 0x02;  // Latch
        PORTA &= ~0x02;
}

void shift_high_bit(){
        PORTA |= 0x01;
}

void shift_low_bit(){
        PORTA &= ~0x01;
}
//shift
void shift_pulse(uint8_t reg_num){
        switch(reg_num){
                case 0:
                        PORTA |= 0x04;
                        PORTA &= ~0x04;
                        break;
                case 1:
                        PORTA |= 0x08;
                        PORTA &= ~0x08;
                        break;
                case 2:
                        PORTA |= 0x10;
                        PORTA &= ~0x10;
                        break;
                case 3:
                        PORTA |= 0x20;
                        PORTA &= ~0x20;
                        break;
                case 4:
                        PORTA |= 0x40;
                        PORTA &= ~0x40;
                        break;
```

```
                case 5:
                        PORTA |= 0x80;
                        PORTA &= ~0x80;
                        break;
                case 6:
                        PORTC |= 0x01;
                        PORTC &= ~0x01;
                        break;
                case 7:
                        PORTC |= 0x02;
                        PORTC &= ~0x02;
                        break;
        }
}
// shift
void shift_write(uint8_t data,uint8_t reg_num){
        int i;
        PORTA |= 0x02;

        for(i=0;i<8;i++)
        {
                if(data & 0b10000000)
                {
                        shift_high_bit();
                }else{
                        shift_low_bit();
                }
                shift_pulse(reg_num);
                data=data<<1;
        }
        //shift_latch();
        PORTA &= ~0x02;
}
```

```
//gear motor
void board_set_front(){
        PORTD |= (1<<PD2);
        PORTD &= ~(1<<PD3);
        //while(!(PIND & (1<<PD0)));
        _delay_ms(5000);
        PORTD &= ~(1<<PD2);
}
//gear motor
void board_set_back(){
        PORTD |= (1<<PD3);
        PORTD &= ~(1<<PD2);
        //while(!(PIND & (1<<PD1)));
        _delay_ms(5000);
        PORTD &= ~(1<<PD3);
}
//servo motor
void board_rotate_front(){
        for(int i=0;i<12;i++){
                PORTC |= (1<<PC2);
                _delay_us(1700);
                PORTC &= ~(1<<PC2);
                _delay_us(18300);
        }
}
//servo motor
void board_rotate_back(){
        for(int i=0;i<12;i++){
                PORTC |= (1<<PC2);
                _delay_us(900);
                PORTC &= ~(1<<PC2);
                _delay_us(19100);
        }
}
```

```
ISR (INT0_vect){
        if (i==1)
        {
                TCCR1B=0;
                pulse=TCNT1;
                TCNT1=0;
                i=0;
        }
        if (i==0)
        {
                TCCR1B|=(1<<CS10);
                i=1;
        }
}
```

6.0 Further work

Including more units which can detect other common eye diseases. And this should be further developed in order to cater the needs of the medical field.

7.0 Individual contribution

Name of the student – **M.A.M. Bandara-174015X**

As a group member and group leader I involved in this project in various ways.

I searched and selected necessary components and bought them. I designed the project and fixed the gear and servo motor to the moving part and connected them with microcontroller. I fixed led bulbs to the vision board and fixed it to moving part. I also connected the LCD, keypad and vision board to the microcontroller. Finally, I admire my teammates' dedication and hard work to succeed in project

Name of the student – **G.C. Sarujan-174149K**

As a group member for this project I helped to fix hardware parts and buy components from electronic shop.

I have some knowledge in fixing electronic components and make connection. In our project I helped to fix vision board led bulbs, gear motor with motor driver and servo motor and this connection.

As a group member I take part with our project from start to now. Finally, I admire my teammates' dedication and hard work to succeed in project

Name of the student – **T.H.K. Sewwandi-174154V**

I searched about the topics, Long vision which is one of the main parts of the house hold vision system. By having instructions from the technicians and nurses I met,planned a relevant system to do the distant vision test . I went to markets to get the necessary components for the long vision part as well as the linear motion part, and searched the most suitable components in the market,I helped to make the structure and did fix the linear motion mechanism to it. Searched for the video which is the basis for the currently using distant vision testing system. Got decisions to use the same standardized long vision number board which is currently using in hospitals. And decided to use a mirror to keep the distance of 6m between the patient and board with the acknowledgement given from one of our members.

I learnt about the atmega32A pin diagram and to use Atmel studio , afterwards I learnt the proteous and designed the pcb to the shift register circuit and attached all the relevant wires to the pcb .I did program the long vision test and created the frame which is able to replace the long vision board and helped to do the connections of wires .

I decided to choose the 74hc595 shift registers in order to connect and control more bulbs at once as Pins in atmega32A is not enough. When it comes to the coding it was done to evaluate row by row. i didn't give a exact time to shift the number as some patients get more time for it .Finally, I admire my teammates' dedication and hard work to succeed in project.

Name of the student – **S. Thayana -174172A**

It's a pleasure to join my group because I had lot of knowledge about Atmega 32A,limit switch and LCD display. I have some knowledge in fixing electronic components and make connection. In our project I helped to fix vision board led bulbs, and limit switches. Connection with motor. I wrote the limit switch functionality. Further I learnt to use proteous software and designed the circuit for the project. I studied about adobe flash and make animation for project model. I prepared the circuit of LCD display and interfacing it with the Atmega 32A Finally, I admire my teammates' dedication and hard work to succeed in project.

Name of the student – **G.S. Alahakoon-174008E**

Learnt about microcontrollers and about using Atmel studio. I searched about vision tests and their procedures. I learnt about how to interface a key pad with a microcontroller. And I learnt to code about how to do operations with a keypad. Keypad was used to enter the patient inputs. I supported in making the colour blindness chart. I also supported in building the circuits. searching components.Did circuit soldering pcb board and vision board back light. Helped to make main wiring and arrange those in proper way. Finally, I admire my teammates' dedication and hard work to succeed in project.

10.0 Appendixes

10.1.References

www.atmega.com

www.electrosome.com

https://youtu.be/kMwy06mAV5U

https://www.reuters.com/article/us-fda-ai-approval/u-s-fda-approves-ai-device-to-detect-diabetic-eye-disease-idUSKBN1HI2LC