Color Detection Project Report

Submitted

In partial fulfillment for the award of the degree of

Bachelors in Computer Application

(with specialization in Data Science)

Dr Sagar Mal Juniwal's

APEX UNIVERSITY

JAIPUR

Supervisor-
Rishank Sharma
Assistant Proffesor

Submitted by-
Rahul Sharma
Enrollment No.:

Department of Computer Science
Apex University
Apex School of Computer & IT

June-2022

Certificate

This is to certify that the project work under titled as

Color Detection Project

is the bonafide work of

Rahul Sharma(………………….)

During this project submission as a partial fulfillment of the Requirement for the Bachelor of computer application IV semester, of the Apex University, Jaipur

------------------	----------------------
Supervisor Dept.	H.O.D of computer

ACKNOWLEDGEMENT

We have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. We would like to extend my sincere thanks to all of them.

We are highly indebted to Mr.Rishank Sharma for his guidance and constant supervision. Also, we are highly thankful to them for providing necessary information regarding the project & also for their support in completing the project.

We are extremely indebted to Dr. KD Gupta , HOD, Department of Computer Science and Engineering, Mrs. Megha Yadav , Project Coordinator, Department of Computer Science and Engineering, for their valuable suggestions and constant support throughout our project tenure.

We also express gratitude towards our parents for their kind cooperation and encouragement which helped me in completion of this project. Our thanks and appreciation also go to our friends in developing the project and all the people who have willingly helped me out with their abilities.

Rahul Sharma
BCA(AU)

TABLE OF CONTENT

LIST OF TABLES

List Of Figures

Title	Page

Abstract

Color detection is the process of detecting name of the color. Here this is easy task for human to detect the color and choose one. But computer cannot detect the color easily. This is tough task for computer to detect the color easily. So that's why we choose this project. Many of the project and research papers are written on this problem. But we use different techniques for this project. Pandas and openCV libraries used in python languages. Open Source Computer Vision Library. Open CV was designed for computational efficiency and with a robust specialise in real-time platform that gives video and audio encoding infrastructure.

1. Introduction

The task of the project was to develop a device which can detect different colors by using the sensor... of the company. The target group of the device are blind persons or persons who can not detect different colors. There are already different systems on the market, but they are very expensive and not useful for affordable.
The new device should be a compact, simple and a cheap solution.

2. Description

The sensor consists of three LED (red, green, blue) two Pin Diodes and a Monitor Diode. Consecutively on Led will be turned on and then the Pin Diodes are measuring the reflection of the light. Each color has a different reflection for the three LED. When the reflection is high the Pin Diodes have a small resistor, is there a little reflection the resistor is high. This value of resistance we transform into a voltage. So we get for each LED a value of voltage which will be converting into the RGB Values. This value were compared with the reference color (a collection of selected colors), and so the device get the right color. The name of the color is given back by a speaker.

a) *The RGB Color Room*

To get an idea how it works to detect the color, we give a short explain of the RGB Color Room.

The RGB Color Room is an additive Model. This means that the three fundamental colors (Red, Green, and Blue) can be added to white. Nearly each color can be described with these three fundamental colors.

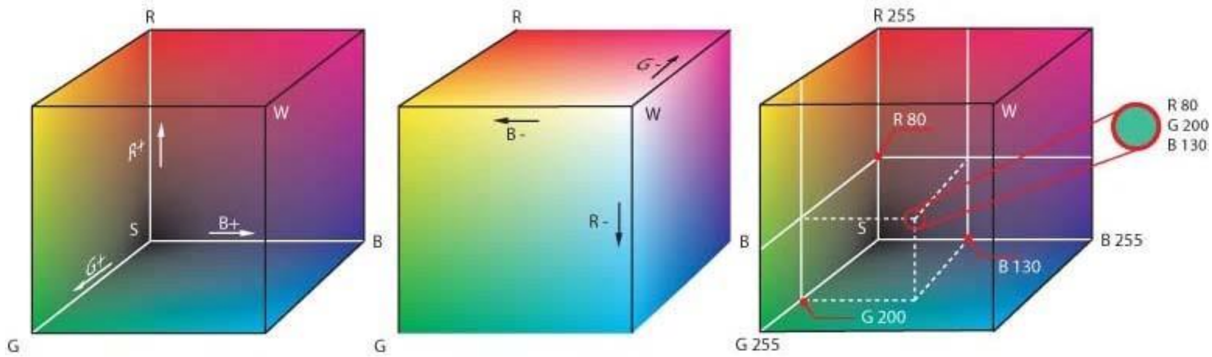


Figure:2.1 ***The RGB Color Room***

In the graph we see the RGB Color Room. On the x-Axes is the color Red, on the y-Axes is the color Blue and on the z-Axes is the color Green.
On the diagonal of the color room we find the grey scale. In the graph we can see that each color is a combination of the three fundamental colors.

The measurements were made, to get an idea, which are the important factors to detect the right color. For the measurements we used a System which contains a Lab VIEW-Program, DAC-Card and the sensor. As Colors we used the „Digital ColorChecker SG"of the company Gretagmacbeth, which consists of the standard colors and much more colors.
In the following table are the colors of the color Checker which we used for our measurements.

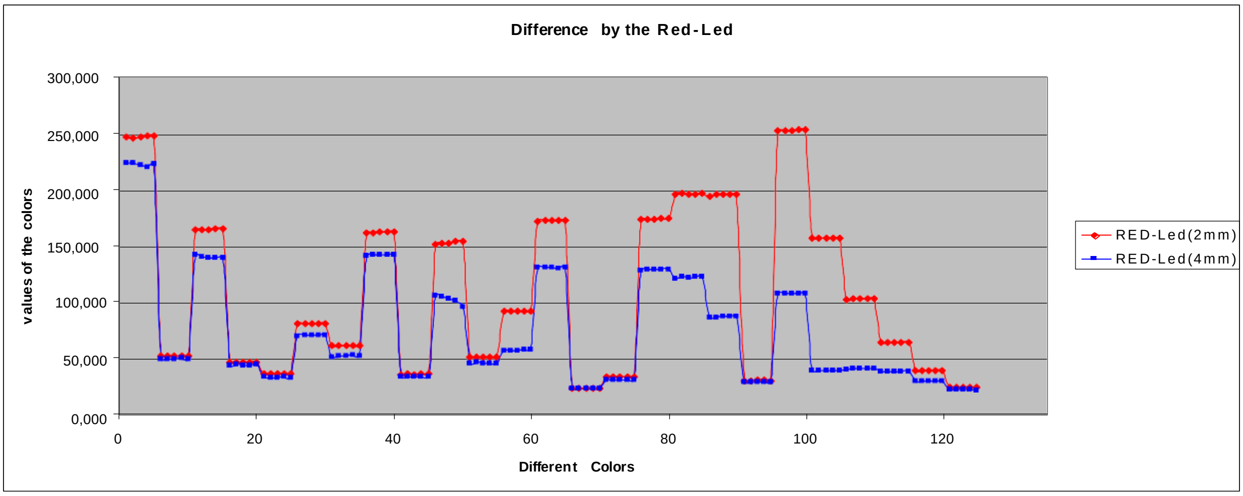
Table 2.1: Koordinaten Table

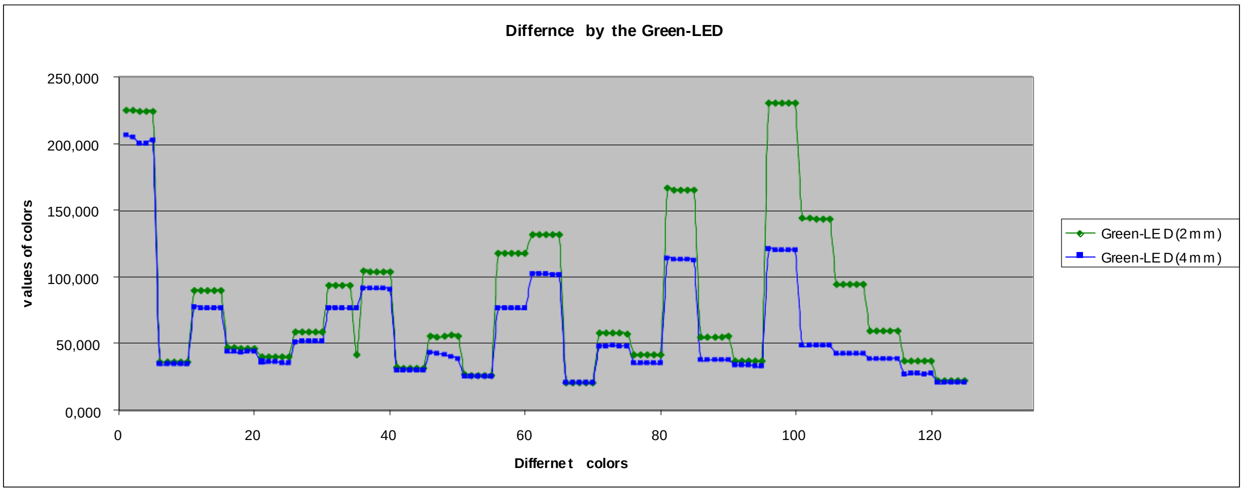
Index	Koordinaten
1	2-E
2	2-F
3	2-G
4	2-H
5	2-I
6	2-J
7	3-E
8	3-F
9	3-G
10	3-H
11	3-I
12	3-J
13	4-E
14	4-F
15	4-G
16	4-H
17	4-I
18	4-J
19	5-E
20	5-F
21	5-G
22	5-H
23	5-I
24	5-J

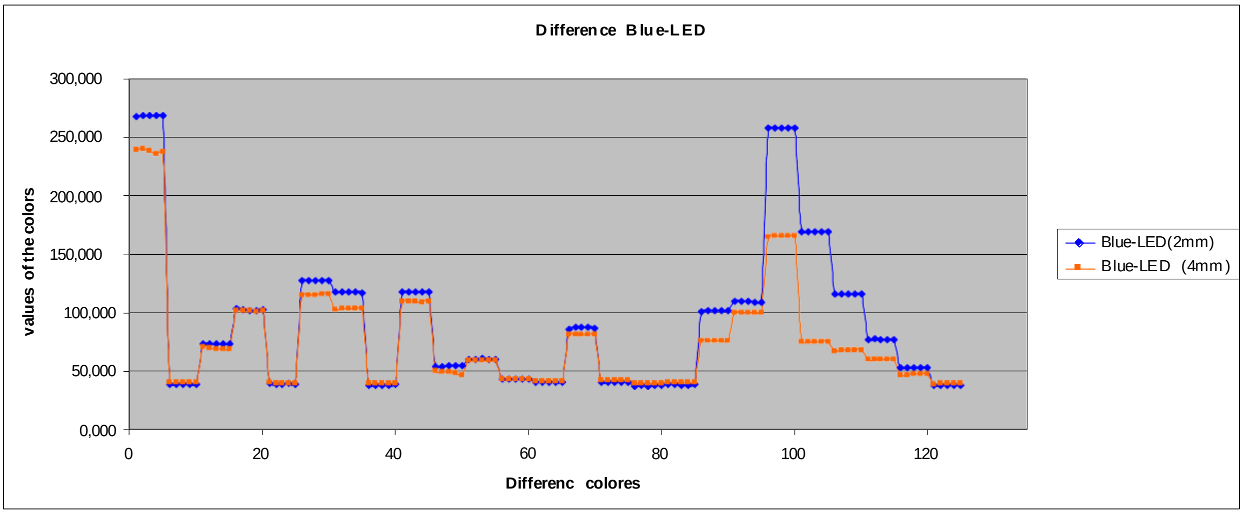
3. Different Measures

a) Measures with different distance to the object

The following measurements were made to detect the influence to the RGB-values, by different distances between sensor and the color.
The distance which was normally 2mm we increased to 4mm.





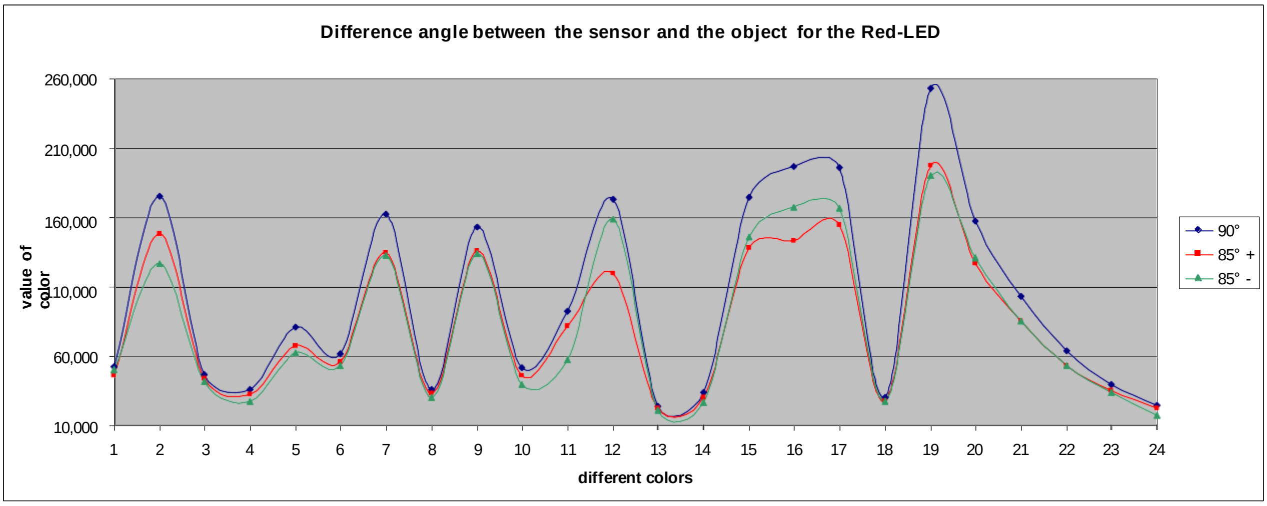


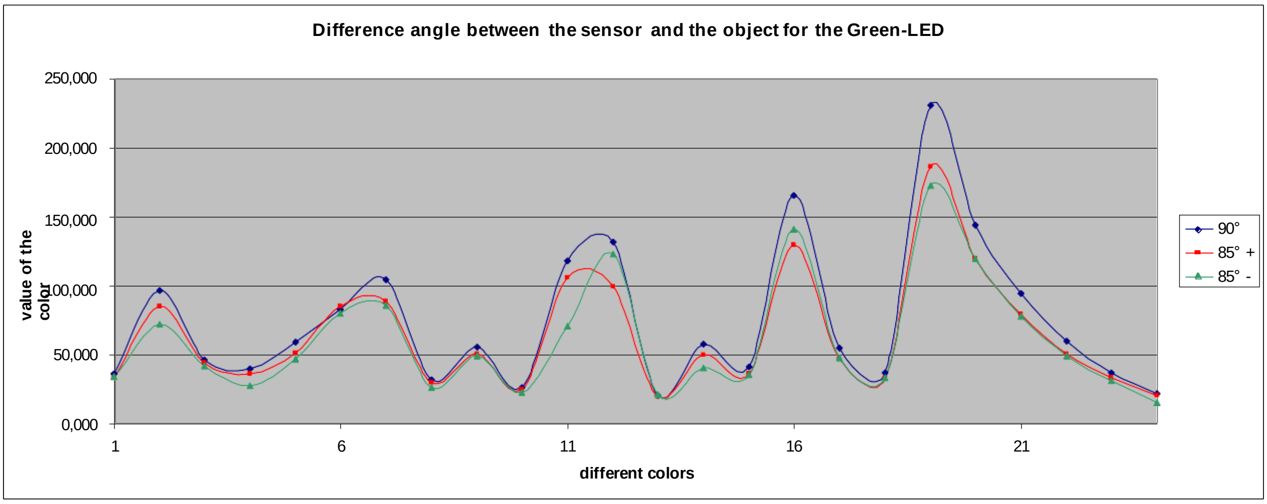
Fgure:3.1 ***Measures with different distance to the object***

The graphs are showing the results by 2mm distance and by 4mm distance for each LED. On the x axes are the 24 standard colors, but for each color we took 5 measurements.
In this diagrams we can see, that the distance to the color, which we want to measure is important, because the values of the three LED are changing because of the distance.

b) Measures with different angel of the sensor

The following measurements were made to detect the influence to the RGB-values, by different Angle between sensor and object. We had fixed the sensor, with an auxiliary construction, in three different angels (85°- / 85°+ / 90°). The following graphs descript the 24 measured colors.





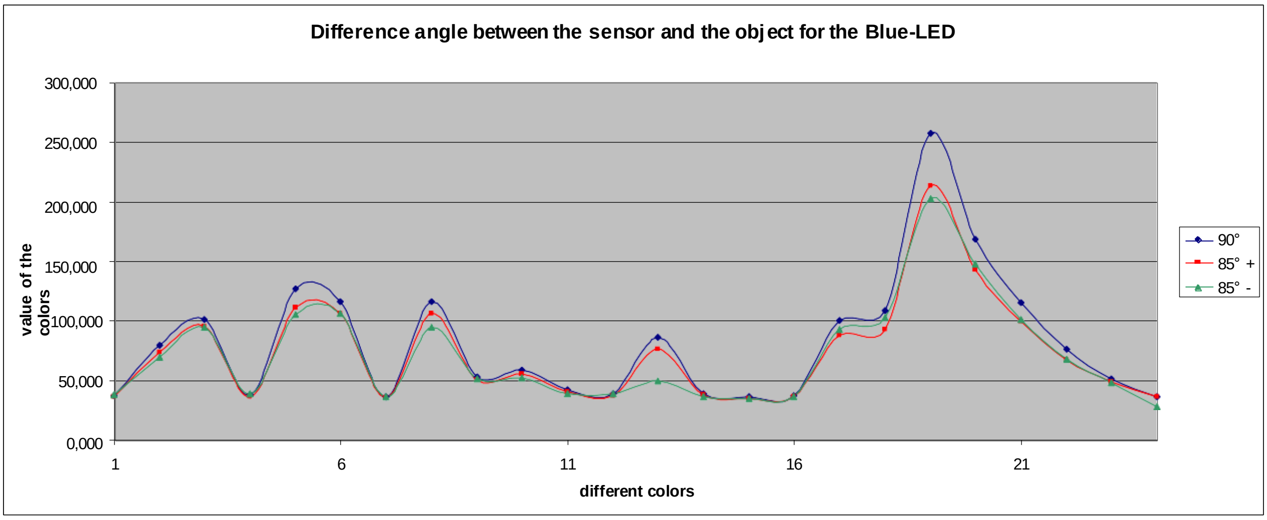


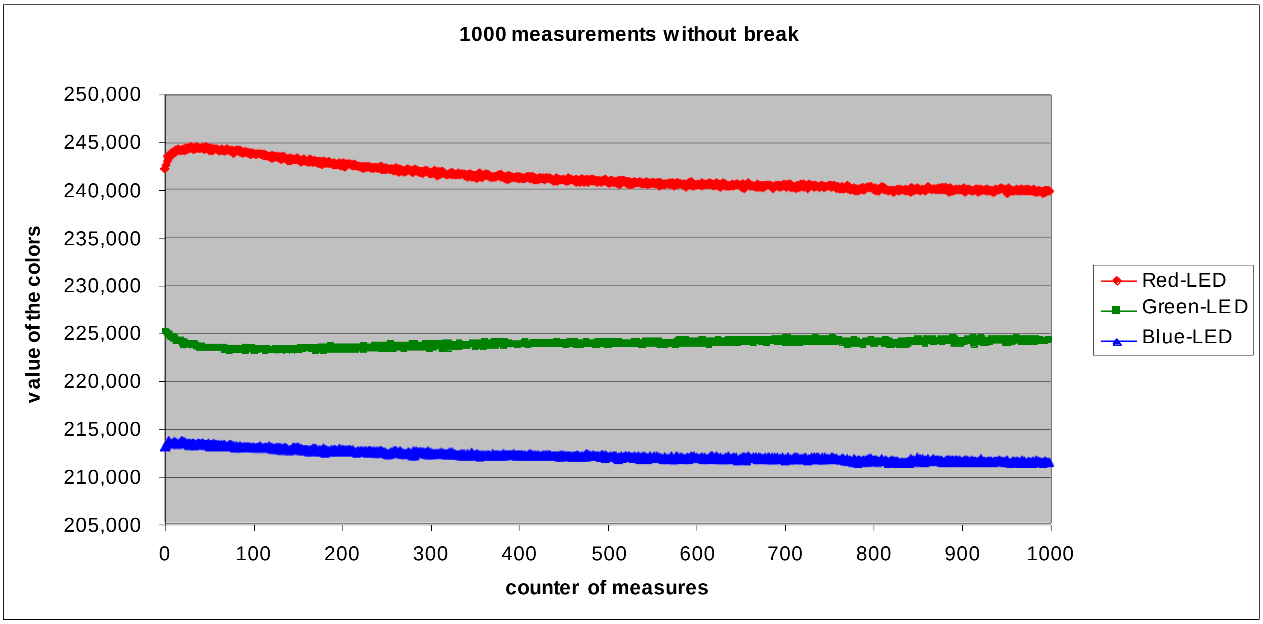
Figure:3.2 ***Measures with different angel of the sensor***

The curve 85° + describes the measurements, when the blue LED is in the origin and the curve 85° - describe the measurements, when the red-LED is in the origin. The curve 90° describes the measurements, when the sensor is justified on the color.
We think, for correct measurement it is important that the angle between the sensor and the object (color) is always the same. So we must show, that the angel is nearly 90°, because it is easier to holt the sensor with a angel of 90° then with 85°+ or 85°-.

c) Temperatures measurement

We had problems because we do not get repeatable measurements. So we made some thoughts about the temperature. We made measurements to find out if a changing of the temperature of the sensor influences a changing of the RGB Values.

So we made 1000 measurements on a white paper without break between the measurements. The results are held on in the following graph.



On this point we can not say that the changing of the values is an effect of the temperature. So we made new measurements.

At first we made 30 measurements and between the measurements we turned all three Leds on for 2 minutes. On the graph below are the results of this measurement.

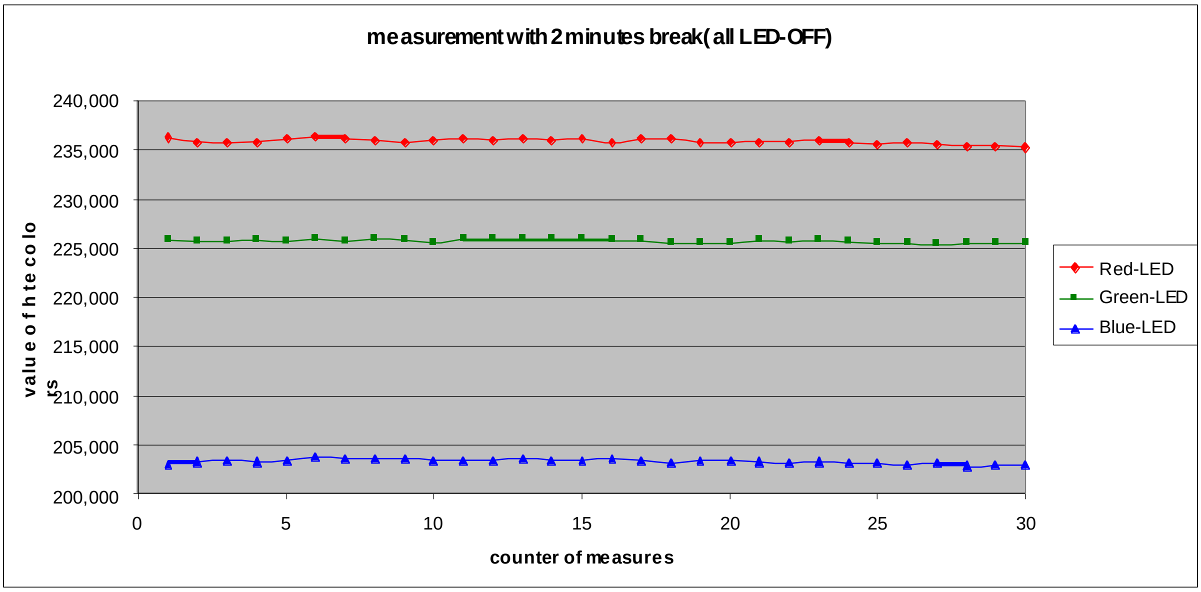


Figure:3.3 ***Temperatures measurement***

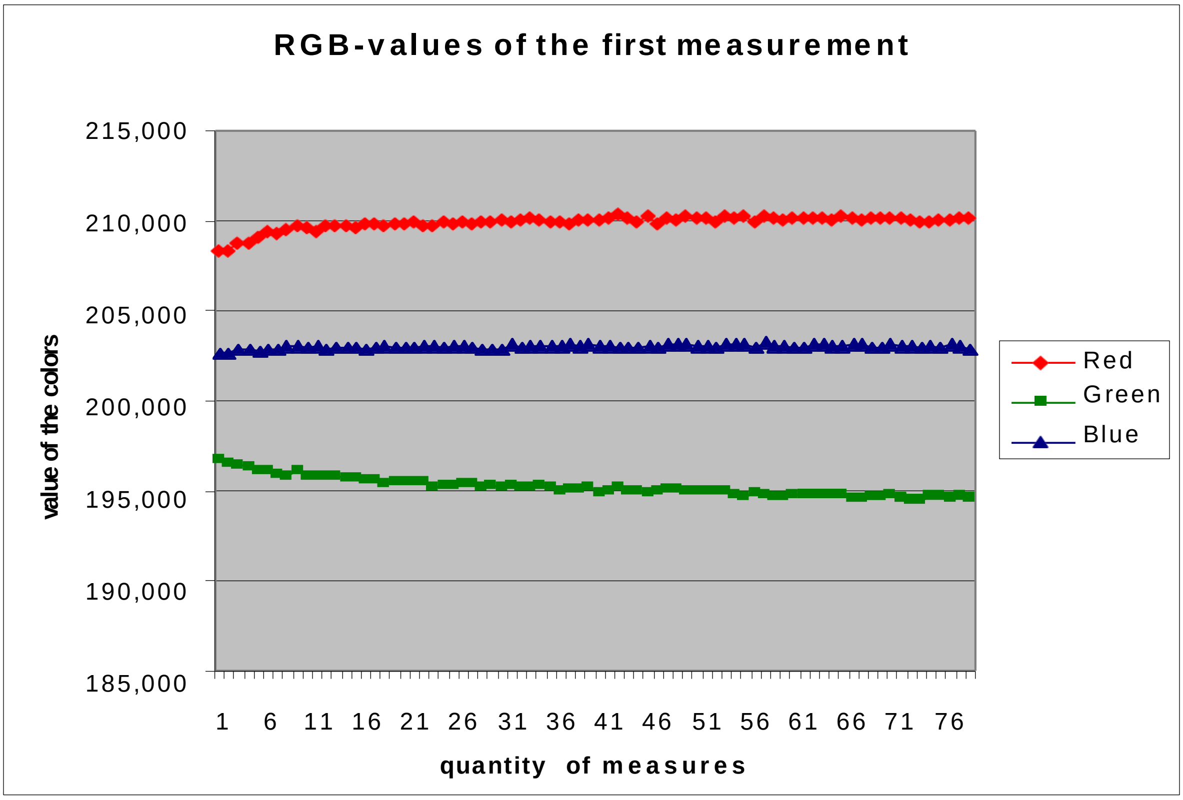
The brake of two minutes causes that the sensor can cool down. The values are nearly constant, so the temperature has an influence to the values rather to the Pin Diodes.

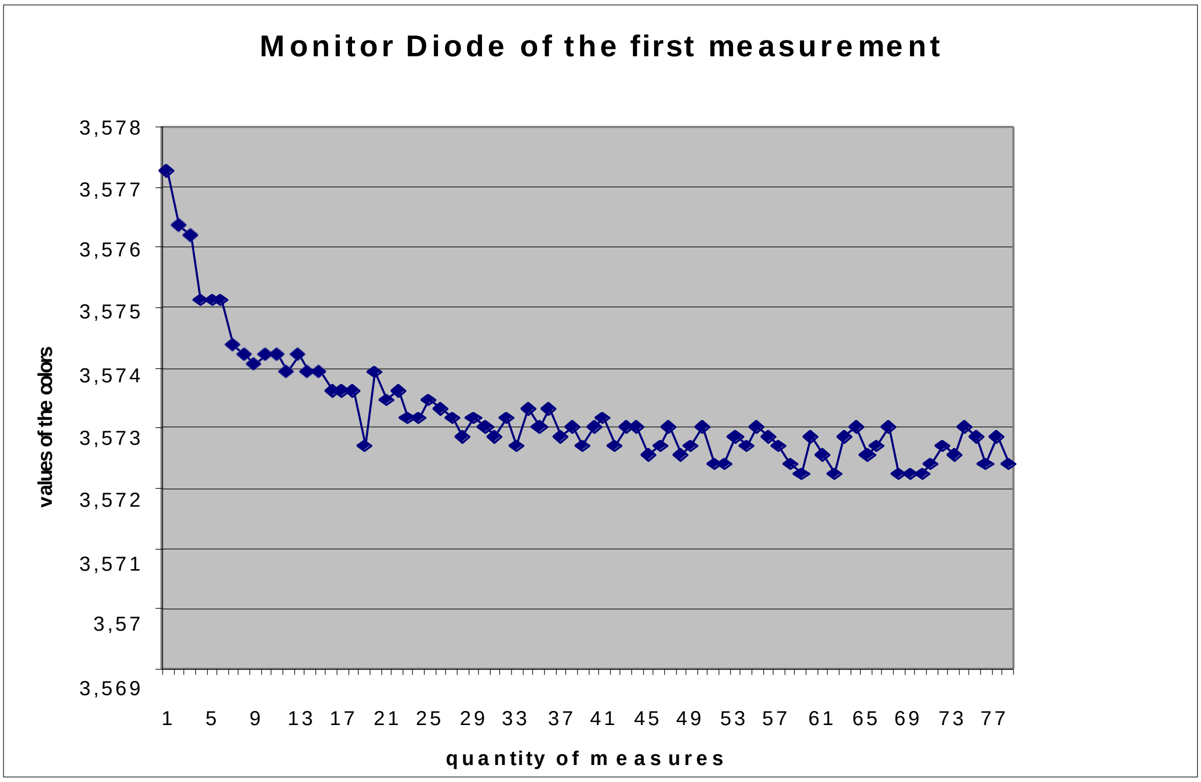
d) Temperatures measurement of the monitor diode

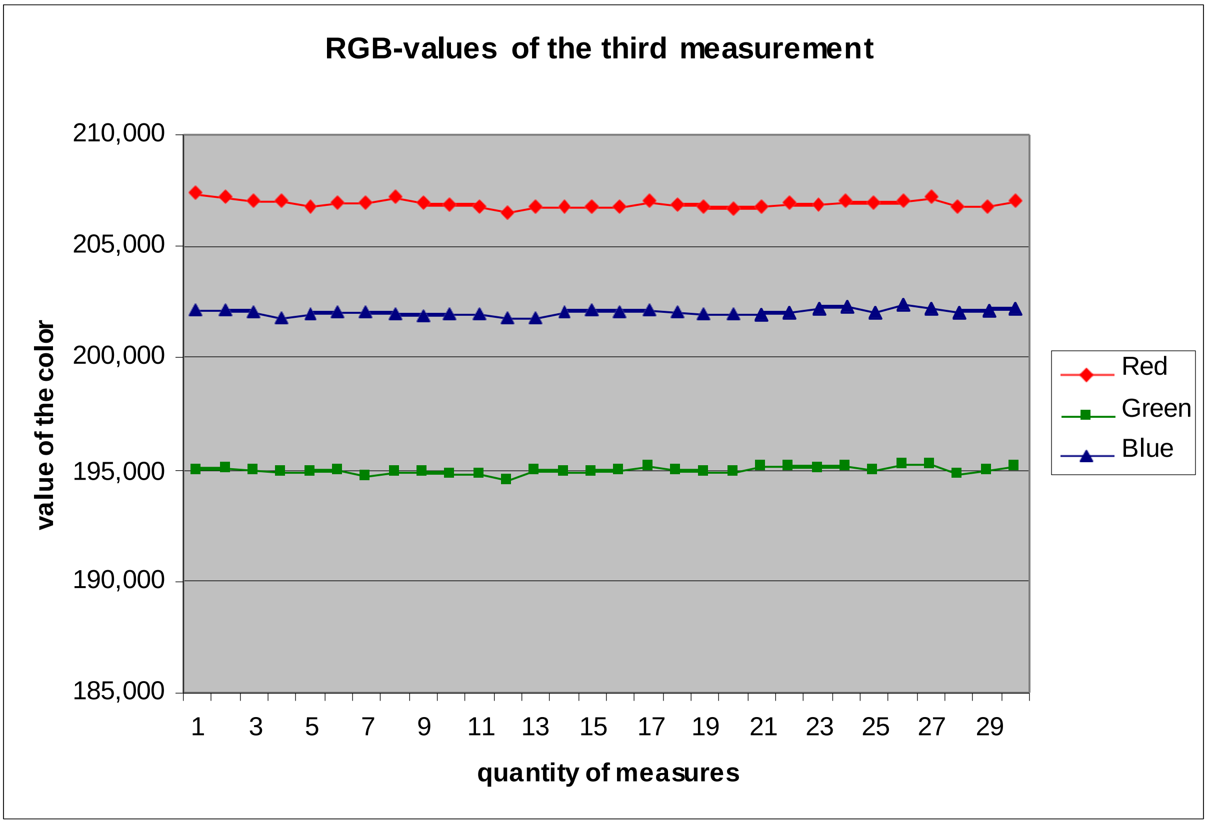
After a consultation with the producer of the sensor we get an idea how we can solve the problem with the temperature. The sensor has inside a Monitor Diode.

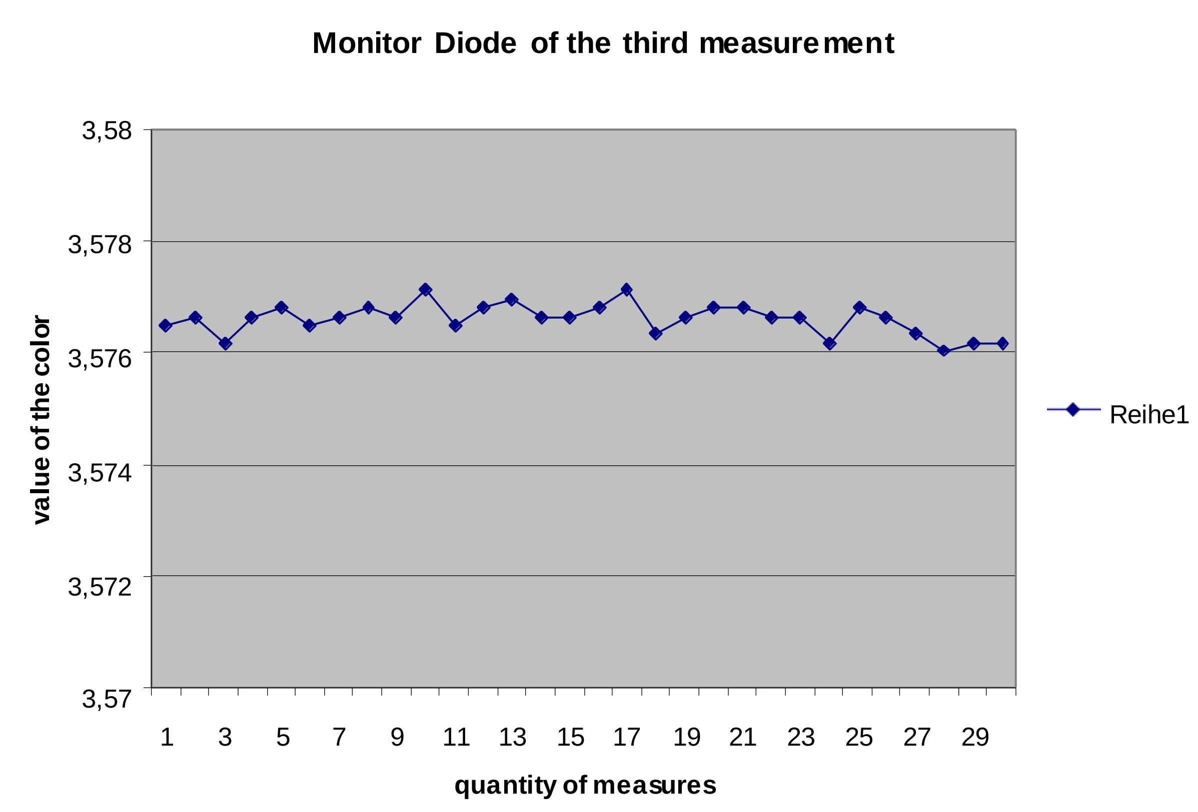
With this Monitor Diode we can measure the temperature, and we can adjust the RGB values.

In the next graphs we made different measurements including the measurements with the Monitor Diode (temperature).






RGB-values of the third measurement
210,000
205,000
200,000
195,000
190,000
185,000
value of the color
1
3
5
7
9
11
13
15
17
19
21
23
25
27
29
quantity of measures
Red
Green
Blue



The voltage of the monitor diode depends on the temperature of the sensor.

Figure:3.4 *Temperatures measurement of the monitor diode*

4. Researches of methods for color identification

An important point is to give out the right color, but at first we have to detect it. To find the right color we use a table with the RGB values of the 24 most important colors.

To explain the methods for detecting the color we take random values for the measured RGB values.

For example:

	R	G	B
measured value	35	100	70

This example shows you the color bluish green with the index 6. The next step is to calculate the difference between the measured values and the reference values which is pictured below.

Table 4.1.Difference between the measured values and the reference values

	Reference value				Difference			
	R	G	B	Index	R	G	B	Index
dark skin	43,009	26,224	15,589	1	8,009	73,776	54,411	1
light skin	147,376	75,147	55,27	2	112,376	24,853	14,73	2
blue sky	36,181	39,887	85,321	3	1,181	60,113	15,321	3
foliage	27,294	33,864	15,758	4	7,706	66,136	54,242	4
blue flower	70,648	49,911	109,701	5	35,648	50,089	39,701	5
bluish green	49,265	101,815	98,12	6	14,265	1,815	28,12	6
orange	148,275	86,562	13,905	7	113,275	13,438	56,095	7
purplish blue	22,561	21,964	102,041	8	12,439	78,036	32,041	8
moderate red	150,229	28,028	33,018	9	115,229	71,972	36,982	9
purple	35,495	13,254	35,623	10	0,495	86,746	34,377	10
yellow green	79,791	129,208	19,224	11	44,791	29,208	50,776	11
orange yellow	164,194	133,479	16,922	12	129,194	33,479	53,078	12
blue	10,26	9,849	79,661	13	24,74	90,151	9,661	13
green	23,519	58,385	16,796	14	11,481	41,615	53,204	14
red	166,618	14,802	11,86	15	131,618	85,198	58,14	15
yellow	194,183	177,114	15,151	16	159,183	77,114	54,849	16
magenta	174,562	27,858	78,173	17	139,562	72,142	8,173	17
cyan	18,601	29,882	90,129	18	16,399	70,118	20,129	18
white 9.5 (.05 D)	235,122	233,511	226,297	19	200,122	133,511	156,297	19
neutral 8 (.23 D)	147,211	150,291	150,387	20	112,211	50,291	80,387	20
neutral 6.5 (.44 D)	89,186	91,952	92,484	21	54,186	8,048	22,484	21
neutral 5 (.70 D)	47,088	49,274	49,878	22	12,088	50,726	20,122	22
neutral 3.5 (1.05 D)	21,758	22,841	23,363	23	13,242	77,159	46,637	23
black 2 (1.5 D)	8,09	8,105	8,311	24	26,91	91,895	61,689	24

a) First method

+

Now we take the 12 minimum values of the red-LED and accept the two other values form the two other colors.

Table 4.2. First method

	12 min of red value	accept	accept
Index	R	G	B
10	0,495	86,746	34,377
3	1,181	60,113	15,321
4	7,706	66,136	54,242
1	8,009	73,776	54,411
14	11,481	41,615	53,204
22	12,088	50,726	20,122
8	12,439	78,036	32,041
23	13,242	77,159	46,637
6	14,265	1,815	28,120
18	16,399	70,118	20,129
13	24,740	90,151	9,661
24	26,910	91,895	61,689

In the second step, we pick out the 6 Minimum values of the green-LED from the table on the top.

	accept	6 min. of green value	accept
Index	R	G	B
6	14,265	1,815	28,120
14	11,481	41,615	53,204
22	12,088	50,726	20,122
3	1,181	60,113	15,321
4	7,706	66,136	54,242
18	16,399	70,118	20,129

In the third step, we pick out the 3 minimum values of the blue-LED from the table above.

	accept	accept	3 min. of blue value
Index	R	G	B
3	1,1811808	60,1125408	15,321
22	12,0878272	50,7264448	20,122
18	16,39924672	70,1175488	20,129

Now we made the sum of all values of each index. The result is:0

Index	absolute deviation
3	76,615
22	82,936
18	106,646

The result with the first method is the color **blue sky** with the index 3.

b) Second method

In the second method we used an allowance for the research, in this we choose index 10. This means, that the measured values can have only a difference of +/-10 to the reference value.

In the first step we pick out the values from each LED, where the difference is lower +/- 10 and made the sum of all values of each index

Table 4.3. Second method

Red-LED				sum of the differences
Index	R	G	B	
10	0,495	86,746	34,377	121,617
3	1,181	60,113	15,321	76,615
4	7,706	66,136	54,242	128,084
1	8,009	73,776	54,411	136,195
Green-LED				
Index	R	G	B	
6	14,265	1,815	28,120	44,200
Blue-LED				
Index	R	G	B	
17	139,562	72,142	8,173	219,876
13	24,740	90,151	9,661	124,552

The result with the second method is the color **bluish green** with the index 6.

c) Third method

The third method is nearly the same like the second method. The difference is only, that we don’t make the sum of all differences. We make a weighting of all values from each index.

d) Comparison of the three methods

Table 4.4 Comparison of memory capacity & computing power

Method	memory capacity	computing power
1	Dependent on the accuracy, the smaller the range of tolerance, the smaller is the storage requirement. However it can occur with the fact that the ‘correct’ color is excluded.	This will need surely at few computing power
2	Depend on the allowance, the smaller it is, the smaller is the storage requirement.	This will need more computing power because we use many values which are needed for the result.

3	Depend on the allowance, the smaller it is, the smaller is the storage requirement.	This will need surely at most computing power, because of the weighting of all values from each index.

As a conclusion, we can say, all the three methods have a possibility, to get the false color-name. So we have found an easier and exacting method to find the correct color.

e) Last method for color identification

The method which we are using to detect the color is a mathematical method. With Red, Green and Blue we can create a RGB - Color room, in this room counts the same mathematical rules like in other vector rooms.
To detect the color we calculate the smallest vector to the reference colors. This is a normal vector calculation.

Calculation: smallest vector…v
Reference red value …rv
Reference green value. …gv
Reference blue value …bv
Measured red value …mrv
Measured green value …mgv
Measured blue value …mbv

$$v = \sqrt{(rv - mrv)^2 + (gv - mgv)^2 + (bv - mbv)^2}$$

For every reference color we get a result, but the smallest value gives as the correct color.

5. Selection of the construction units - Hardware

a) Introduction

The Hardware is an important component of this project. It is divided in three mean parts: Microcontroller, Sensor, Speech part. Each mean part needs other parts and components for the correct functionality.

b) Power supply

A 3 Volt Battery is used for the power supply. The Microcontroller and the Speech Chip needs 5 Volt, so the 3 Volt is not enough. The MAX756 is a step up converter which converts an input voltage between 1.7 Volt to 5 Volt up to 3 Volt or to 5 Volt. This Chip detects also when the battery is low. With 5 Volt output voltage it is possible to get an output current up to 200 mA.

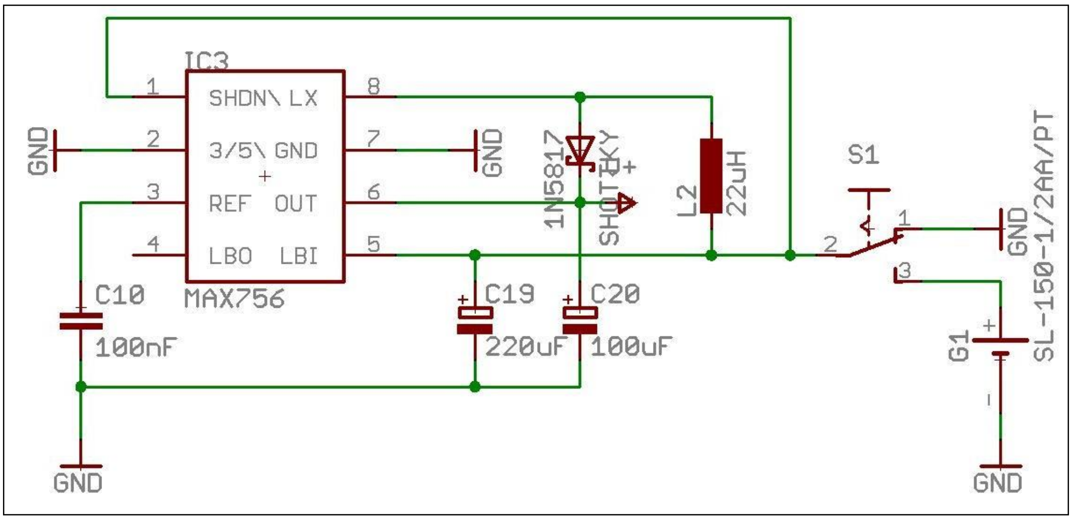


Figure:5.1 Power supply

The picture shows the Layout of the step up converter.

c) Sensor

The Sensor consists of a Red LED, Green LED, Blue LED, two Photo Diodes, and a Monitor Diode. There is also a connected ground and a reference Voltage for the Photo Diodes.

The Red LED needs a voltage around 1.91 Volt and a current of 14mA, so the resister between microcontroller and the LED must be 22K. The blue LED needs a voltage of 3.75 Volt and a current of 7mA so the resister must be 180Ohms. The Green LED takes 5 Volt and a current of 32mA, but the microcontroller has an output current of a maximum of 25mA. So the current is given by a transistor. The basis of this transistor is connected to the microcontroller and the emitter is connected to the green LED. If

the output port of the microcontroller is set low the transistor locks. If the output port is high the green LED gets the current.
The Photo Diodes are used for detecting the light intensity. The resistor or the conductivity of the Photo Diodes is changing when the light intensity is changing.
The Monitor Diode is used for detecting the temperature in the sensor. It is also changing the resistor when the temperature is changing.

d) The Amplifier

An amplifier is used to amplify the signal of the Photodiodes. The amplifier is an inverting amplifier. It consists of an OPV LM358 a resistor and a capacitor. If the resistor of the Photo Diodes is changing the output voltage of the OPV is also changing. The capacitor is used to stable the voltage.

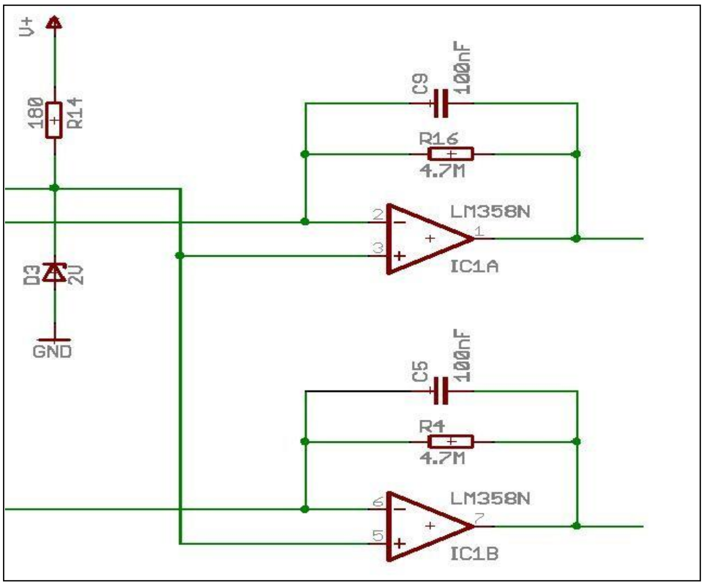


Figure:5.2 Amplifier

e) Microcontroller

We use the PIC18F2455 from microchip. More information about the microcontroller please visit the datasheet (http://www.microchip.com/downloads/en/DeviceDoc/39632c.pdf).
The Voltage of the Pin Diode1 is connected with the Pin 2 (Port A0), and the Pin Diode 2 is connected with Pin 3 (Port A1). On the Pin3 we find the Monitor Diode. On the Pins 11, 12, 13 are connected the Leds (Blue, Red, Green). It is possible to detect the low battery with the Port RC5 (Pin16). On Pin 18 (Port RC7) is the bush bottom to start a measurement. The Pins 21, 22, 23, 24, 25 (Port B0…B4) are the address lines for the Speech Chip. With Pin 27 (Port B6) we can detect when a played message is finish. To start playing a message we give the signal on Pin 28 (Port B7).

f) Speech chip

We are using the speech chip to give out the name of the color by a speaker. After researches we have decided to use an ISD chip, because it was the only chip which is able to give out an analog signal.
For us it was important because the digital output of a voice is very hard to understand.

We are using the ISD 1420, which can store 20 seconds. The following graph shows the protective circuit.

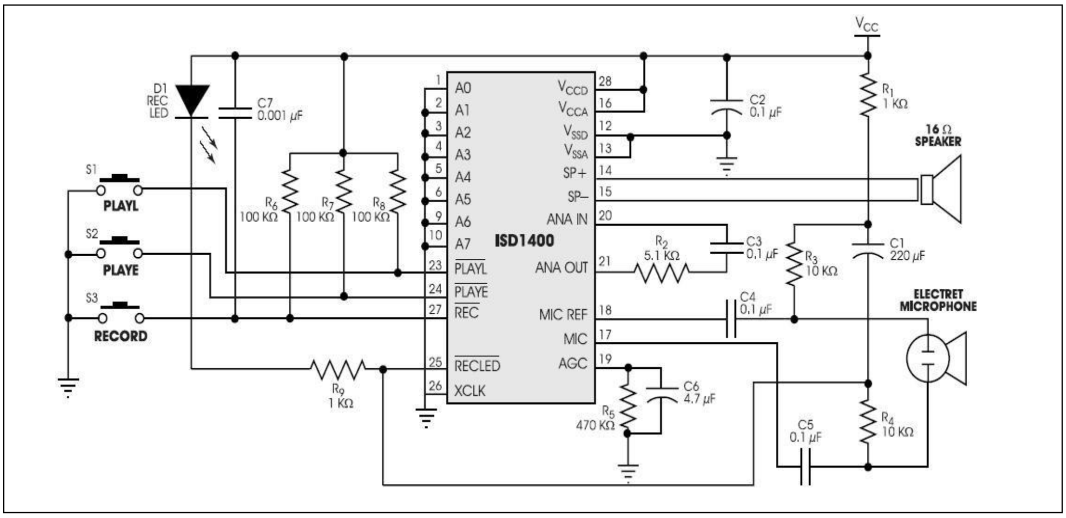


Figure:5.3 Speech chip

This circuit was important to record the different messages (color names). The addresses A0-A7 we are using to select the position of the message with a binary code. The buttons PLAYL / PLAYE / REC are to play or store the messages. The AGC input is for amplify of the input signal. The input signal can be transmitted by two different ways. The first way we are using is a Microphone and the second way we set the output “ANA OUT” after the capacitor at Ground. Then we connect an analog signal (PC/ MP3-player …) on the “ANA IN”.
For the output we are using an 8 Ohm speaker.

g) Schematic

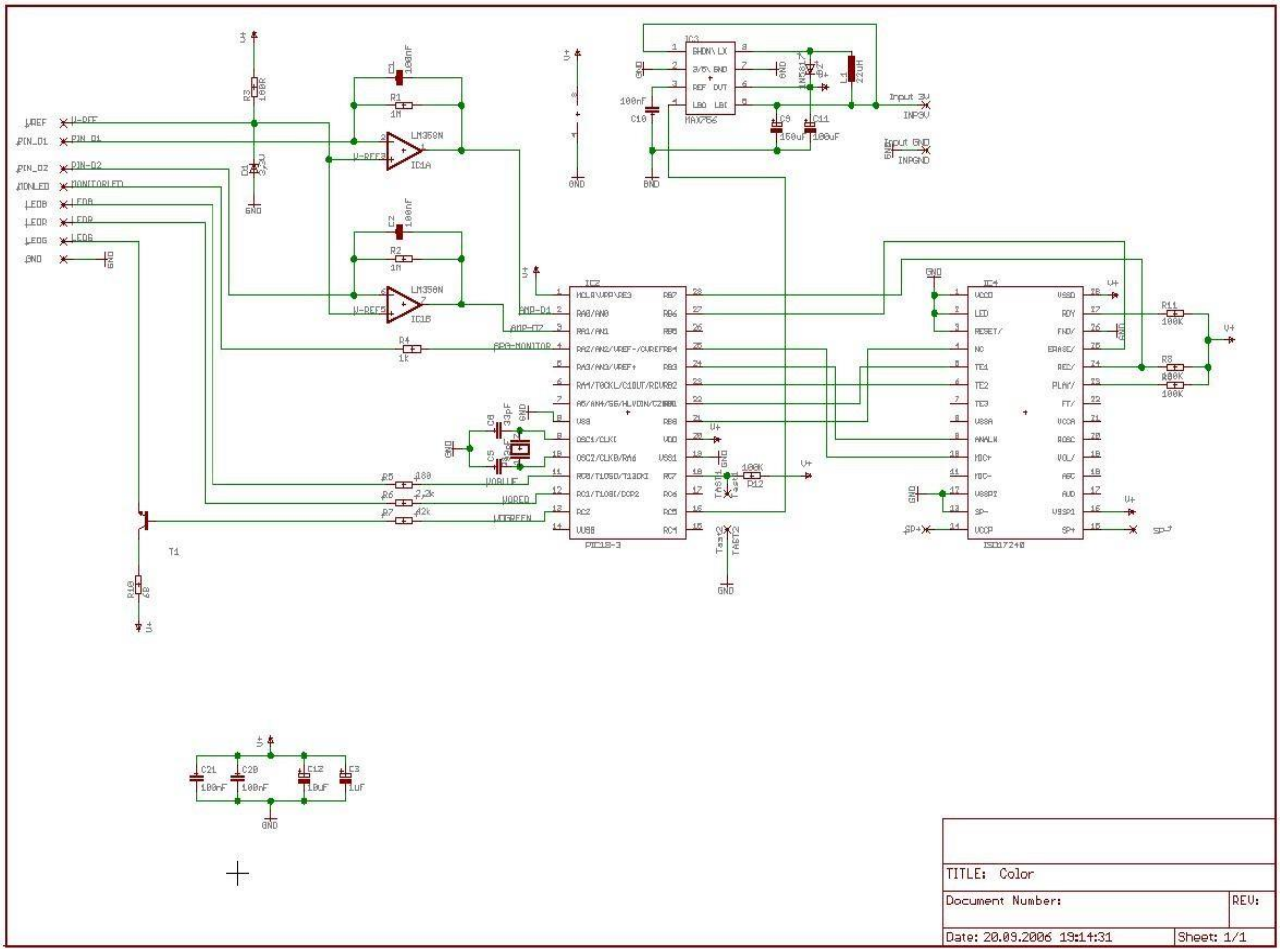


Figure:5.4 Schematic

h) Board

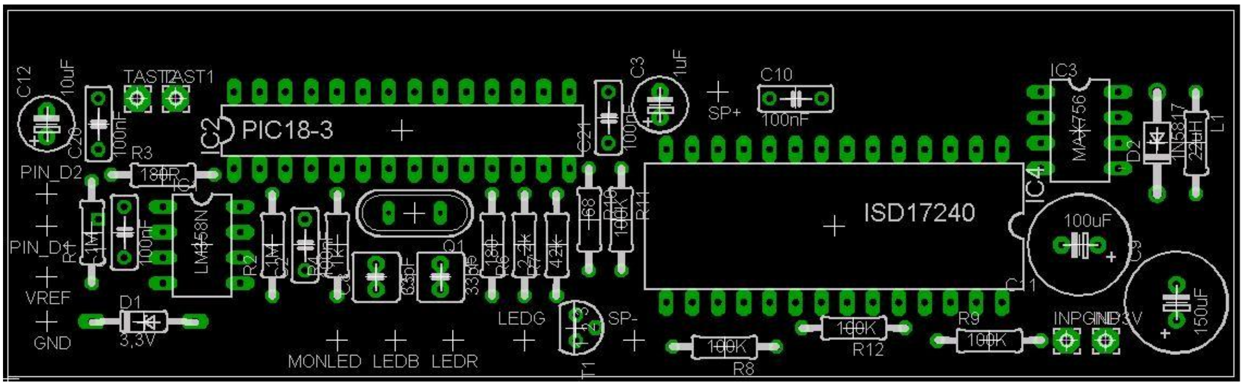


Figure:5.5 Amplifier

6. <u>The design</u>

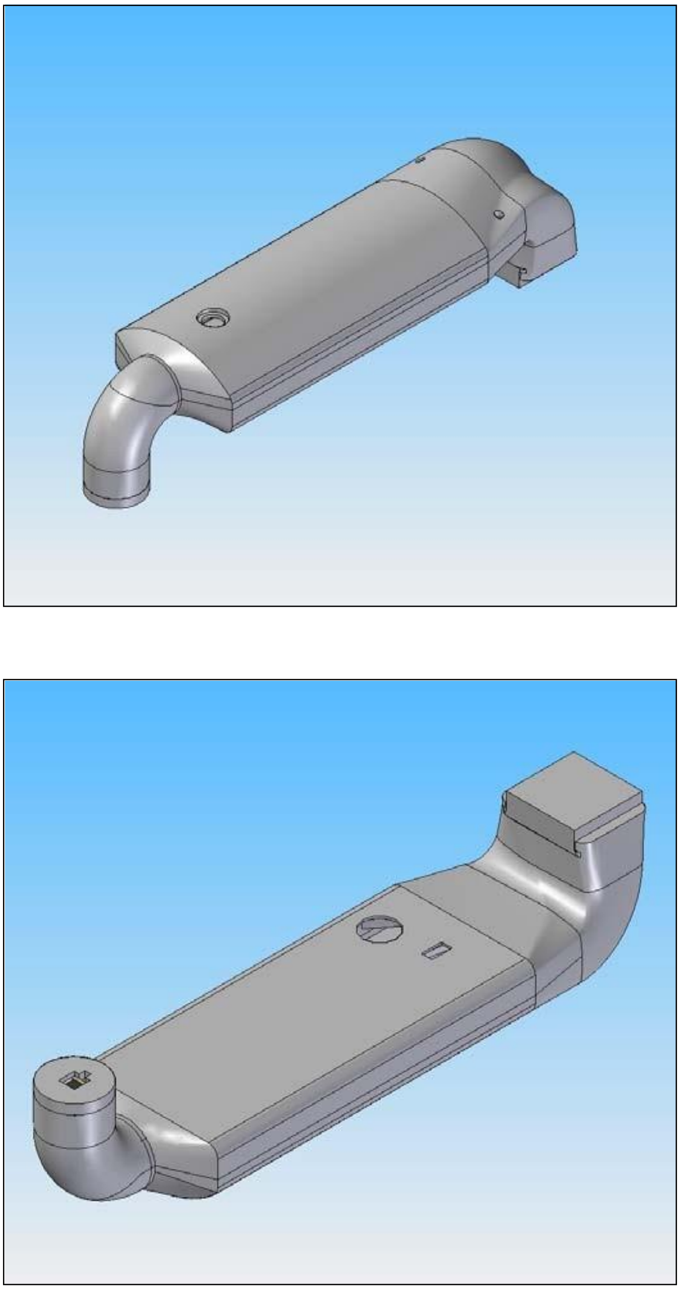


Figure:6.1 The design

7. Introduction of Software Development

The goal of this report is to development of software for microcontroller to develop the embedded system of the color detection project. The basic idea of the software is to communicate with two modules of the embedded system sensor and speech chip.

First of all, Ports configuration done of microcontroller. After that it turns on each LED of the sensor in sequence and when the Led is turning on, Pin diodes of the sensor measures the reflection of Led light and it was simply done by calling analog to digital function (ADC).

Then some mathematical operations are come in the picture, which are applied on result of ADC to make it comparable with universal RGB values (stored in arrays).

Next step is to compare this measured and calculated RGB values with universal RGB values and here, software finds the minimum distance between measured and universal RGB values.

At the end, after finding minimum distance it is ready to send 5 bits address to speech chip where the colours are stored and after receiving this message speech chip is responsible to speak the matching color.

The microcontroller used was PIC18F2455 from microchip. The programmer tools were MPLAB and PICSTART Plus from Microchip. All the programming development was made in ANSI C using Microchip C18 compiler.

For more information complete software implementation is included.

References

[1] R. O. Duda, P. E. Hart and D. G. Stork. Pattern Classification. John Wiley and Sons Inc., 2001.
[2] G. Gomez. On selecting colour components for skin detection. In ICPR02, pages II: 961–964, 2002.
[3] A. Pentland M. Turk. Eigenfaces for recognition. Journal of Cognitive Neuroscience, Volume 3, Number 1, pages 71–86, 1991.
[4] J.M.Rehg M.J. Jones. Statistical color models with application to skin detection. Int. J. of Computer Vision, Volume 46, Number 1, pages 81–96, 2002.

a) The Program

```
import cv2
import numpy as np
import pandas as pd
import argparse

#Creating argument parser to take image path from command line
ap = argparse.ArgumentParser()
ap.add_argument('-i', '--image', required=True, help="Image
Path")
args = vars(ap.parse_args())
img_path = args['image']

#Reading the image with opencv
img = cv2.imread(img_path)

#declaring global variables (are used later on)
clicked = False
r = g = b = xpos = ypos = 0

#Reading csv file with pandas and giving names to each column
index=["color","color_name","hex","R","G","B"]
csv = pd.read_csv('colors.csv', names=index, header=None)

#function to calculate minimum distance from all colors and get
the most matching color
def getColorName(R,G,B):
    minimum = 10000
    for i in range(len(csv)):
        d = abs(R- int(csv.loc[i,"R"])) + abs(G-
int(csv.loc[i,"G"]))+ abs(B- int(csv.loc[i,"B"]))
        if(d<=minimum):
            minimum = d
            cname = csv.loc[i,"color_name"]
    return cname
```

More