

April 11<sup>th</sup>, 2005

Lucky One  
School of Engineering Science  
Simon Fraser University  
Burnaby, British Columbia  
V5A 1S6

RE: ENSC 440 Post-Mortem for a Colour Identifying Device

Dear Mr. One:

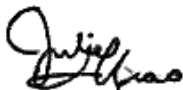
The enclosed document, *Post-Mortem for a Colour Identifying Device—ColourIt*, explains the methods of design and choices in implementation.

Our project is to develop a handheld device, ColourIt that can determine the colour of a point. ColourIt can determine the values of red, green and blue a specific colour is comprised of. ColourIt is ideal for designers, painters, hobbyists, as well as the colour vision deficient.

The purpose of this post-mortem is to describe the technical issues we have encountered and explain deviations from the original design specifications. This document will also discuss changes in schedules and budgets. Future improvements to the project will also be covered.

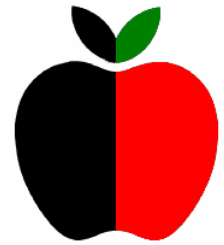
ColourSense Inc. is comprised of three innovative, diligent and ambitious fourth year engineering students – Julie Chao, Linda Ni and Bill Yang. I am more than willing to answer any questions or concerns in regards to our functional specification. Please feel free to contact me by phone at 604-764-2608 or by email at [coloursense-ensc440@sfu.ca](mailto:coloursense-ensc440@sfu.ca).

Sincerely,



Julie Chao  
Chief Executive Officer  
ColourSense Inc.

Enclosure: *Post-Mortem for a Colour Identifying Device—ColourIt*



**ColourSense Inc.**

*Post-Mortem for a  
Colour Identifying Device*

# ColourIt

*Team Members:*

Julie Chao  
Linda Ni  
Bill Yang

ctchao@sfu.ca  
lni@sfu.ca  
billy@sfu.ca

*Submitted to:*

Lucky One  
Amir Niroumand  
Mike Sjoerdsma  
Tony Ottaviani  
Scott Logie  
Steve Whitmore

lucky@sfu.ca  
amniroum@sfu.ca  
msjoerds@sfu.ca  
tottavia@sfu.ca  
slogie@sfu.ca  
whitmore@cs.sfu.ca

*Issued Date:* April 14<sup>th</sup>, 2005

## Executive Summary

In John Locke's famous essay of 1690, he pondered whether the "*violet produced in one man's mind by his eyes were the same that a marigold produced in another man's, and vice versa*[1]." The colours perceived by an individual would not be the same as the colours perceived by another at a different time or different place. While philosophers and scientists have debated this issue for centuries, the engineers of ColourSense have endeavoured to resolve it through the development of a portable colour-identifying device - ColourIt.

ColourIt is a handy device capable of determining the values of red, green and blue a specific colour is comprised of. ColourIt is precise, portable, affordable and user-friendly.

The unending list of applications for ColourIt includes colour distinction for the colour vision deficient, exact colour matching and verification for painters and designers, determining color differences between digital prints and the original item, copyright protection of images and detection of art fraud.

The post-mortem outlines the design and functionality of our final product. This report also describes the technical issues we have encountered and explain deviations from the original design specifications. In addition, we will discuss future improvements to ColourIt, provided more resources are available.

This document will compare and justify the differences between the projected costs with the incurred costs, as well as the proposed project schedule with the actual timeline. The individual contribution of each team member will also be acknowledged at the end of the document.

## Table of Contents

Executive Summary ..... ii

Glossary ..... 1

1. Introduction ..... 2

    1.1. Scope ..... 2

    1.2. Intended Audience ..... 2

2. Present State of the System ..... 3

    2.1. Colour-Sensing Unit ..... 3

    2.2. Central Control Unit ..... 4

    2.3. User Interface Unit ..... 4

    2.4. Packaging ..... 4

    2.5. Power Specifications ..... 4

3. Deviation from Proposed System ..... 5

    3.1. Colour-Sensing Unit ..... 5

    3.2. Central Control Unit ..... 6

        3.2.1. Addition of Black Reference ..... 6

        3.2.2. User-Defined Black and White Reference Option ..... 6

    3.3. User Interface Unit ..... 6

    3.4. Packaging ..... 7

    3.5. Power Specifications ..... 8

4. Future Improvement of System ..... 9

    4.1. Colour-Sensing Unit ..... 9

    4.2. Central Control Unit ..... 10

    4.3. User Interface Unit ..... 10

    4.4. Packaging ..... 10

5. Assessment of Project Management ..... 11

    5.1. Deviations from Scheduling ..... 11

    5.2. Deviations from Budget ..... 12

6.	Interpersonal and Technical Experiences .....	14
6.1.	<i>Julie Chao (Chief Executive Officer)</i> .....	14
6.2.	<i>Linda Ni (Project Manager &amp; Technical Specialist)</i> .....	14
6.3.	<i>Bill Yang (Chief Technology Officer)</i> .....	14
7.	Conclusion .....	15
8.	References .....	16
	Appendix A: Software Flow Charts .....	17
	Appendix B: Instruction Set of PIC18F452 .....	17

**List of Figures**

FIGURE 1: SYSTEM BLOCK DIAGRAM..... 3  
FIGURE 2: FINAL PACKAGING OF SYSTEM ..... 4  
FIGURE 3: ORIGINAL PLACEMENT OF SENSOR..... 5  
FIGURE 4: NEW PLACEMENT OF SENSOR ..... 5  
FIGURE 5: FIRST PROTOTYPE DESIGN..... 7  
FIGURE 6: SECOND PROTOTYPE DESIGN ..... 7  
FIGURE 7: E986D-CTN BEFORE MODIFICATION ..... 8  
FIGURE 8: COLOURCHECKER CHART ..... 9

**List of Tables**

TABLE 1: ACTUAL PROJECT TASKS AND SCHEDULE..... 11  
TABLE 2: COST FOR FINAL PROTOTYPE..... 12  
TABLE 3: PROJECTION OF ONE-TIME COST FOR FINAL PRODUCT ..... 13  
TABLE 4: RECURRING COST FOR FINAL PRODUCT ..... 13

## Glossary

<b>EEPROM</b>	nonvolatile memory where writing to memory is complicated and safeguarded so data stored in that section is not easily overwritten
<b>IR</b>	infra-red
<b>LCD</b>	Liquid Crystal Display
<b>LED</b>	Light Emitting Diode
<b>RGB</b>	Red, green, blue
<b>UDATA</b>	user defines variables and allocate a byte to each variable

## **1. Introduction**

For the past thirteen weeks, ColourSense Inc. has been devoted in the research and development of ColourIt – a colour-identifying device. When used to detect the colour of an area, ColourIt informs the user the specific values of red, green and blue the colour is comprised of via a LCD screen. Since ColourIt is precise, portable, affordable and user friendly, it is intended for both amateurs and professionals in industries involving exact colour matching. Our handy device will also help those with colour vision deficiency.

ColourIt is ready for immediate use, although it also offers the user the convenience of calibration on different surfaces or materials. For example, a contractor needs matching indigo tiles while a decorator needs matching indigo drapes. But ColourIt is originally calibrated on paper, not tiles or drapes. Hence, individual calibration for the contractor and decorator will give them more desirable results. This is the key feature that separates ColourIt from other colour-identifying devices in the industry

### **1.1. Scope**

This post-mortem is a detailed description of the development of a colour-identifying device. The overall system specifications include the colour-sensing unit, the central control unit and the user interface unit. The purpose of this post-mortem is to describe the technical issues we have encountered and explain deviations from the original design specifications. This document will also discuss changes in schedules and budgets we have made from our original proposal. Future improvements to the project will also be covered. The individual contribution of each team member will also be acknowledged at the end of the document.

### **1.2. Intended Audience**

The post-mortem is primarily intended for project managers to evaluate the progress of ColourSense Inc. The project manager will use this document to determine whether project objectives were met and make decisions in regards to future expansions of ColourIt.



## 2. Present State of the System

The development of ColourIt is split up into three units: colour sensing, central control and user interface. The most current block diagram of our system is shown in Figure 1.

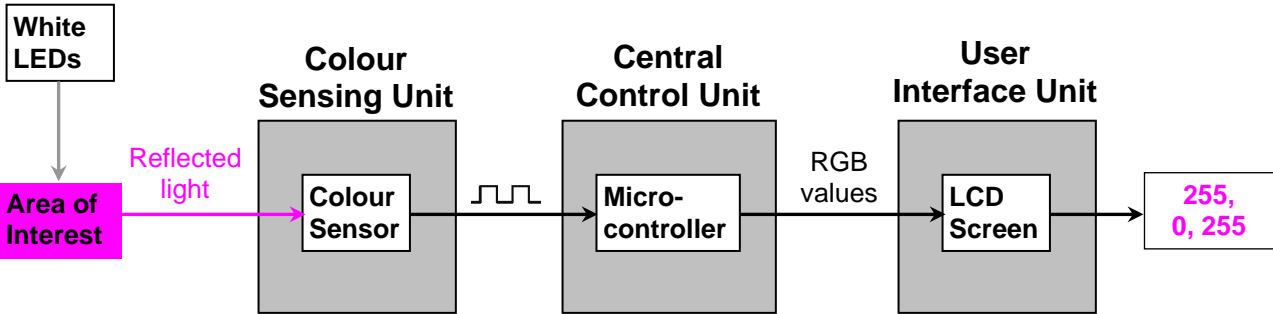


Figure 1: System Block Diagram

The system is powered by one standard 9V battery. ColourIt has a power switch and two buttons for the user: sample and toggle. Once on, ColourIt can be used immediately to determine the RGB values of the colour of any flat surface over 4cm<sup>2</sup>. The user needs to place the head (the colour sensing area) over the surface of interest and click the sample button. Immediately, the LCD screen will display the RGB values for the user.

The user can also decide to turn off/on the white LEDs (contained in the head) by clicking the toggle button. This allows the user to determine the colour of areas that have its own source of illumination.

Some users may require precise colour matching on glossier or rougher surfaces. Although they get consistent results when they sample the colour, the output RGB values will appear to be off because different surfaces produce different reflectivity. Hence, such users have an option of conducting their own calibration using their set of materials considered “pure white” and “pure black”. First, after they turn on the device, they need to hold the sample button until the LCD screen displays, “WhiteRef Done”. Then, they need to hold the toggle button until the LCD screen displays, “BlackRef Done”. Now, the system is calibrated for this set of materials.

### 2.1. Colour-Sensing Unit

In colour sensing, it is imperative to have consistent colour detection. Hence, the device is capable of isolating external light and providing its own light source to illuminate the area of interest. The TCS230 from TAOS is the colour sensor used to pick up the light intensity based on the colour of the area. One single output channel is sent to the microcontroller to be processed for the levels of red, green, blue and intensity.

## 2.2. Central Control Unit

The central control unit is a bridge between the colour sensing unit and user interface unit. This middle unit revolves the use of control software to handle the microcontroller. The microcontroller is capable of powering and controlling the input signals to the sensor. The oscillatory clock on the microcontroller is fast enough to receive input from the sensor and able to check the status of two buttons continuously. The data collected from the colour-sensing unit is processed and prepared to output in the user interface unit.

## 2.3. User Interface Unit

The user interface unit integrates the use of the LCD module with the microcontroller and involves a power switch, sample button and toggle button. After switching the microcontroller on, the user scans the region of interest and the RGB values of the colour will be displayed on the LCD screen. The toggle button allows the user to turn the white LEDs on/off. The user also has an option of re-defining white and black reference.

## 2.4. Packaging

Figure 2 shows the final packaging of the system uses a PVC conduit body (E986D-CTN). One of the openings is removed and the other is used to house the colour-sensing unit. PVC insulates the internal components and is easy to cut openings for the LCD screen, switch and buttons. The final size of ColourIt is about 77×27×26 mm (not including the head).



Figure 2: Final Packaging of System

## 2.5. Power Specifications

The current system requires 5.0V and a load current of 60mA. Therefore, a 9V lithium battery is used with the LM341 voltage regulator to obtain a continuous 5.0V output.

### 3. Deviation from Proposed System

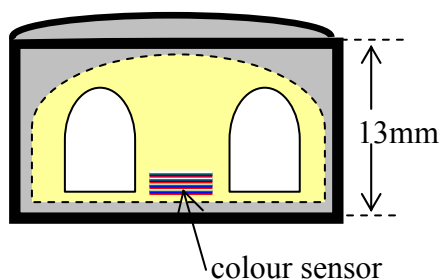
Although the project was carefully planned, some of the firmware and hardware did not turn out as initially anticipated. Therefore, the functionality of the device had to be refined and the design of the system had to be slightly modified.

The major change in functionality is that ColourIt only provides the user with the RGB values of the colour sampled and not the exact name of the colour. This section will provide more detailed explanations of this change in feature. In brief, we have experienced many unexpected spectral errors with the system that made produced RGB combinations which did not match any set of predefined colours. We also realized that people interested in precise colour-matching will only be concerned with the RGB values; the names of colours are irrelevant in their applications. Therefore, our new proposal was to develop a colour-sensing device that was precise and produced consistent RGB readings of a colour.

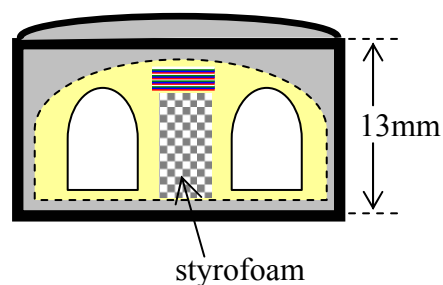
#### 3.1. Colour-Sensing Unit

The arrangement of the white LEDs around the colour sensor remained the same. The only change in physical design involved placing the sensor as physically close to the protection plastic cover as possible. Before, when the sensor was positioned at the far back of the colour-sensing head (as shown in Figure 3), the readings from the sensor was greatly interfered by the colour of the inner wall in the protection tube.

To avoid interference of colour readings, styrofoam was used to support the sensor so that it is positioned as close to the top as possible (as shown in Figure 4).



**Figure 3: Original Placement of Sensor**



**Figure 4: New Placement of Sensor**

Another modification in the colour-sensing unit allows the user to disable the four white LEDs by clicking on the toggle button. This option is desirable in situations where the user wants to determine the RGB values in an illuminated region of interest like halogen light bulbs or the light from a coloured projector.

## 3.2. Central Control Unit

### 3.2.1. Addition of Black Reference

ColourIt was originally required to identify colours based on a white reference to a purely white surface in the proposal. Through rigorous testing, we have found that TCS230 responds to a purely white surface and sends out a high frequency in all channels. As colours dim on its intensity scale, the sensor outputs a lower frequency. However, the sensor does not respond a zero frequency for a purely black surface. For having more accurate results, we realized there was a need to subtract the relatively low frequency corresponding to the black colour in each channel. Therefore, an 8-bit pixel number of a specific colour channel,  $y$ , is calculated from the frequency output for that colour channel,  $x$ , in the following way:

$$y = \frac{x - \text{Black Ref}}{\text{White Ref} - \text{Black Ref}} \times 255$$

### 3.2.2. User-Defined Black and White Reference Option

Amount of light does not reflect uniformly from different types of materials with the same colour. Sensitivity of the colour sensor, TCS230, varies to surfaces of different materials. Cloth may reflect less light back to the sensor than a piece of glossy paper in the same colour. ColourIt may output different readings due to discrepancies in reflectivity. The original design was to perform a one-time white calibration prior to production; the users were not required to adjust the reference for pure white. In order to produce a more accurate reading taken the material of the surface into consideration, we have supplemented an option for user to define both white and black references in their applications.

## 3.3. User Interface Unit

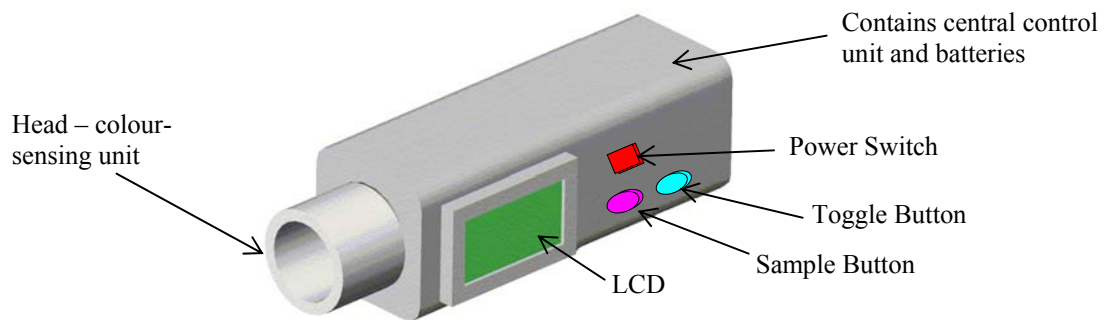
One deviation from the original proposal is that the user will not be provided with the name of the colour sampled, only the RGB values. Another change is that the toggle button is not used to switch from the name of the colour to the RGB values. Instead, the toggle button is to turn the white LEDs on or off.

The final ColourIt does not sample colours in real-time. Therefore, the user does not have to hold down the sample button to sample the colour. The user simply places the head of ColourIt over the region of interest and clicks the sample button to determine the RGB values of a specific colour. Once a colour is sampled and its RGB values will remain until the sample button is pressed again. This gives the user the time to place the device down and record the RGB values before the next sample is taken.

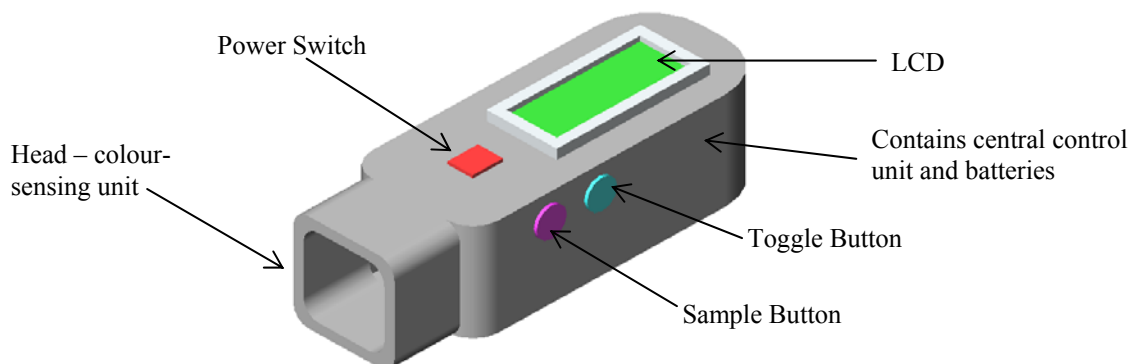
The user can press the sample button for more than two seconds (until the LCD screen displays, "WhiteRef Done") to conduct white calibration. This procedure can be repeated on the toggle button to conduct black calibration ("BlackRef Done").

### 3.4. Packaging

There were many ideas on how to package the setup. We had to decide whether to construct the setup from scratch, hire professionals to machine our design or modify objects with similar shapes to match our design. AutoCAD was used to give two realistic views of our final product (shown in Figure 5 and Figure 6). The advantage of the second prototype design was that the head was not circular; it is easier to work with square-shaped PCBs.



**Figure 5: First Prototype Design**



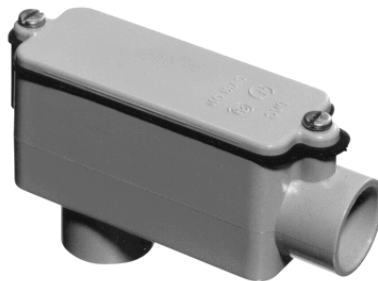
**Figure 6: Second Prototype Design**

With the time and funding permitted, it was better to modify an existing product. After failure with pipes, flashlights and switchboxes, we finally came across PVC conduit bodies. Conduit bodies are usually used to “provide access to the conduit system through a removable cover at a junction of two or more entries [1].”

The E986D-CTN (shown in Figure 7) is the final choice of conduit body and can be easily purchased from Home Depot for less than \$3. This model was chosen for its size, material and overall shape. The E986D-CTN is just barely too small to hold the microcontroller, batteries, sensor, white LEDs and LCD module. Therefore, 2mm was milled off of its original thickness of 4mm to provide just enough room. The overall dimension is 77×27×26 mm (not including the head), which is just small enough so that it can fit into most pockets or purses.

The PVC material will avoid problems of shorts when the PC board touches the conduit body. PVC is water proof and light and easy to cut up for the LCD module, switch and buttons.

There is a lid that can easily be unscrewed to reach the internal components of the setup. This makes it easy for the user to replace the 9V battery. There are two protruding holes just over 20mm. The hole on the side is used to hold the sensor system, while the bottom hole was removed completely.



**Figure 7: E986D-CTN Before Modification**

### **3.5. Power Specifications**

Originally, we desired to use two standard AA batteries or one 3V lithium coin battery with a step-up voltage regulator (the MAX756) to generate 5V. This option was desirable because AA batteries are affordable and easily available. The coin battery, on the other hand, is very small. However, because 3.0V is used to step up to 5.0V, both types of batteries were insufficient to provide a load current of 60mA. Furthermore, implementing the MAX756 involved large capacitors, inductors and rectifiers. These components were more expensive and altogether take up a lot of space.

After some research and experimentation with voltage regulators, the LM341 is used instead. The LM341 requires a minimum of 7.2V to produce a continuous 5.0V output. Therefore, a 9V lithium battery is used to power the system, which is relatively affordable and also easily available for the user to replace. A typical 9V battery is 625mAh, meaning the system can operate approximately 10 hours before the battery needs to be replaced.

## 4. Future Improvement of System

After this stage of design and development, we realize we could further customize ColourIt to fit current market demands for our next stage. This section depicts future improvements and expansions to our system that could be undertaken.

### 4.1. Colour-Sensing Unit

Because our system produced a set of RGB values on colours that did not match any predefined set, we realized that many spectral errors were involved. For example, it was difficult to decide which red can be perceived as pure red with RGB of 255, 0, 0. Near the end of our project, we discovered the ColorChecker Chart from GretagMacbeth for \$130 (as shown in Figure 8).



Figure 8: ColourChecker Chart

The ColorChecker is desirable because it consists of a unique test pattern scientifically designed to help determine the true color balance of any color rendition system. This standard is widely used in photography and publishing because it provides the needed standard with which to compare, measure and analyze differences in color reproduction in various processes. These squares are not only the same color as their counterparts, but also reflect light the same way in all parts of the visible spectrum. In other words, the squares will match the colors of natural objects under any illumination and with any color reproduction process [2]. Therefore, the ColorChecker will definitely be ordered for future improvement of the project. This would be the standard to follow to provide a set of the names of colours.

Another source of spectral error is the focusing of the sensor. Hence, a very specific lens can be placed in front of the sensor to reduce the field of view, so the sensor only sees the colour of the target surface. Therefore, the interfering noise colours will be blocked out to produce a more accurate reading.

Lastly, because the sensor responds to infra-red light in addition to visible colours, and because most surfaces reflect IR of some degree, an IR filter can be placed in front of the sensor to make the reading more accurate.

## 4.2. Central Control Unit

A  $3 \times 3$  correction matrix can be used to minimize the discrepancy of measured RGB values by the sensor and the actual RGB values of a standard. Although the correction matrix was attempted, the system still produced high errors because of the spectral errors mentioned in section 4.1.

The firmware can also be modified so that the user can choose to store the measured RGB values to EEPROM for future use.

The measured RGB values can be calculated and transformed into other colour spaces such as HSL, CMYK, LAB, sRGB. Allowing the user to choose between the most convenient set of colour representation

An additional button can be implemented so when pressed, the illuminating LEDs will turn on, turning ColourIt into a white LED flash light. Flashlights illuminated with white LEDs have many advantages over the conventional incandescent light bulbs: longer battery life, greater shock resistance and better directional angle of light.

## 4.3. User Interface Unit

The power switch currently used is free and taken from broken and abandoned lab equipment.

A LCD module with backlight can be used, so that can be read even in dark environments.

Buttons of different shapes or colours can be used to decrease the chances of pressing the wrong button.

## 4.4. Packaging

If the device is to be mass produced, we will have the time and funding to develop a more ergonomic packaging. The packaging can be custom made by professionals and remanufactured in factories.

The battery is in an isolated compartment so that when the user will not be able to tamper with the internal circuitry when replacing the battery.

The head containing the colour-sensing unit can be also be removed and changed for heads of different sizes so the user can detect colour of varying sizes of area.



## 5. Assessment of Project Management

### 5.1. Deviations from Scheduling

We have not only completed our project on time, but demonstrated project results ahead of schedule. We conducted thorough research and compiled a list of components on demand before placing an order. While waiting for arrival of parts, we designed firmware algorithm and reviewed datasheet of all components. We utilized all labour and resources efficiently so we were able to present more than we expected at the Employer Open House. After the mid-semester demonstration, we have refined our project scope as well as firmware algorithm to enhance performance of ColourIt. We have successfully built a final prototype for final demonstration. The following table details our tasks and timeline.

**Table 1: Actual Project Tasks and Schedule**

Tasks	Timeline
Update budget and schedule	Jan 24- Jan 26
Study ICD2 and reference book for PIC18F452	Jan 24- Jan 30
Compile a list of components and place orders	Jan 24- Jan 30
Design algorithm for firmware	Feb 1- Feb 6
Circuit design for colour sensing unit	Feb 6- Feb 12
Implement algorithm for firmware	Feb 6- Feb 18
Circuit construction for colour sensing unit	Feb 13- Feb 18
Pin assignment	Feb 13- Feb 18
Output experimental results on an LED array	Feb 20- Feb 21
White calibration on colour sensor	Feb 20- Feb 23
Seek for proper voltage regulators	Feb 21- Feb 23
Circuit design and construction for first stage prototype	Feb 22- Feb 23
Reorganize circuit and polish firmware for Employer Open House	Feb 23- Feb 24
Demonstrate functional integration of sensor and central control unit	Feb 24-Feb 24
Employer Open House	Feb 24-Feb 24
Refine project scope	Mar 5- Mar 12
Study LCD module	Mar 14- Mar 18

Study usage of EEPROM	Mar 14- Mar 18
Research details on colour space and vector representation	Mar 14- Mar 18
Procure an accurate colour reference	Mar 20- Mar 24
PCB layout	Mar 20- Mar 24
Implement LCD on the first stage prototype	Mar 20- Mar 30
Package ColourIt	Mar 28- Apr 4
Testing and Debugging	Apr 1 - Apr 4
Research improvement on colour accuracy and matrix correction	Apr 5- Apr 6
Optimize algorithm for pixel calculation with a homemade colour checker	Apr 5- Apr 7
Prepare and practice for presentation	Apr 7- Apr 7
Final demonstration	Apr 8- Apr 8
<b>End of project</b>	

### 5.2. Deviations from Budget

We have spent \$251.26 on all components and necessary resources, which is well within estimation of \$300.58 from our proposal. We have effectively utilized existing resources as well as samples provided from the manufacturers to reduce project expense. Following is a table detailing our total cost for a final prototype.

**Table 2: Cost for Final Prototype**

<b>Parts</b>	<b>Cost</b>
Home-made Colour Checker	\$10.00
Microcontroller (PIC18F452)	\$25.00
In-Circuit-Debugger / Program loader for PIC18F452 (Property of SFU Engineering Science )	\$0
Crystal (10 MHz)	\$7.34
LCD display and driver	\$25.00
RGB sensor(s)	\$45.00
LEDs (white)	\$44.06
IC SOCKET	\$4.04
Serial port connector	\$5.06
Prototyping board(s)	\$15.00

Batteries (9V Alkaline, 3V Lithium)	\$15.00
Light touch buttons	\$0.76
Power switch	\$5.00
Casing (Flashlight, pipes, conduit bodies,)	\$20.00
Laser Pointer	\$5.00
Miscellaneous components	\$25.00
<b>Total</b>	<b>\$251.26</b>

We have also thought about putting ColourIt into market and estimated one-time cost and recurring cost for mass production as in Table 3 and Table 4.

**Table 3: Projection of One-time Cost for Final Product**

<b>Parts</b>	<b>Cost</b>
Colour Checker	\$150.00
License for Copy Right	\$10,000.00
ICD2 / Program loader	\$250.00
Analytic Electrical Equipment	\$1,000.00

**Table 4: Recurring Cost for Final Product**

<b>Parts</b>	<b>Cost</b>
Firmware	\$65
Microcontroller(s)	\$10.00
Crystal (10 MHz)	\$3.67
Voltage regulator	\$2.00
LCD display and driver	\$12.00
RGB sensor(s)	\$4.50
LEDs (white)	\$22.00
Prototyping board(s)	\$5.00
Batteries (9V Alkaline)	\$5.00
Light touch buttons	\$0.76
Power switch	\$2.00
Casing (Plumping pipes)	\$5.00
Miscellaneous components	\$5.00
Assembly labour	\$60
<b>Total</b>	<b>\$199.00</b>

Profit in each unit is estimated to be 30% of the recurring cost, which will be approximately \$60.00. The suggested price of each unit is \$259.00.

## **6. Interpersonal and Technical Experiences**

### **6.1. Julie Chao (Chief Executive Officer)**

Julie started off by defining a specific vision for this project. She shared her vision with her team and together, they have developed a successful prototype.

Julie was largely responsible for the electronic portion of the project – from Digi-Key orders to soldering and wiring of components. The circuit design and implementation involved powering the system and the colour-sensing unit. Julie also worked on a correction matrix with Bill to try to reduce the errors of RGB values. She also worked extensively on the packaging; her goal was to produce a device that is portable, small and user-friendly.

Julie also played a major role in the documentation process.

### **6.2. Linda Ni (Project Manager & Technical Specialist)**

Linda was initially responsible for research on feasible project ideas, schedule and budget determination, task division, and component and resource acquisition along with efficient group meetings.

Linda has had an opportunity to gain knowledge in colour model, colour space, and colour standards defined in different applications. She was also involved in the first stage firmware design and implementation to determine pixel values in each colour channel as well as investigate methods to improve accuracy of ColourIt.

Linda also learned success in project is not only about brilliant ideas but also about efficient execution while maintaining team dynamics.

### **6.3. Bill Yang (Chief Technology Officer)**

As the Chief Technical Office of Colour Sense Inc., Bill's main contribution to this project was the development of the firmware. The firmware was programmed in assembly under the Integrated Development Environment provided by Microchip Inc. The ICD2 debugger from Microchip Inc. was also used as an assistance to speed up the firmware development of ColourIt.

Bill was also responsible for the algorithm in converting frequency to RGB values, which involved extensive experimentation and testing of the colour sensor. He also worked with Julie reducing error using the correction matrix. In addition, Bill has conducted extensive research throughout the project and contributed to various documentations of the project.

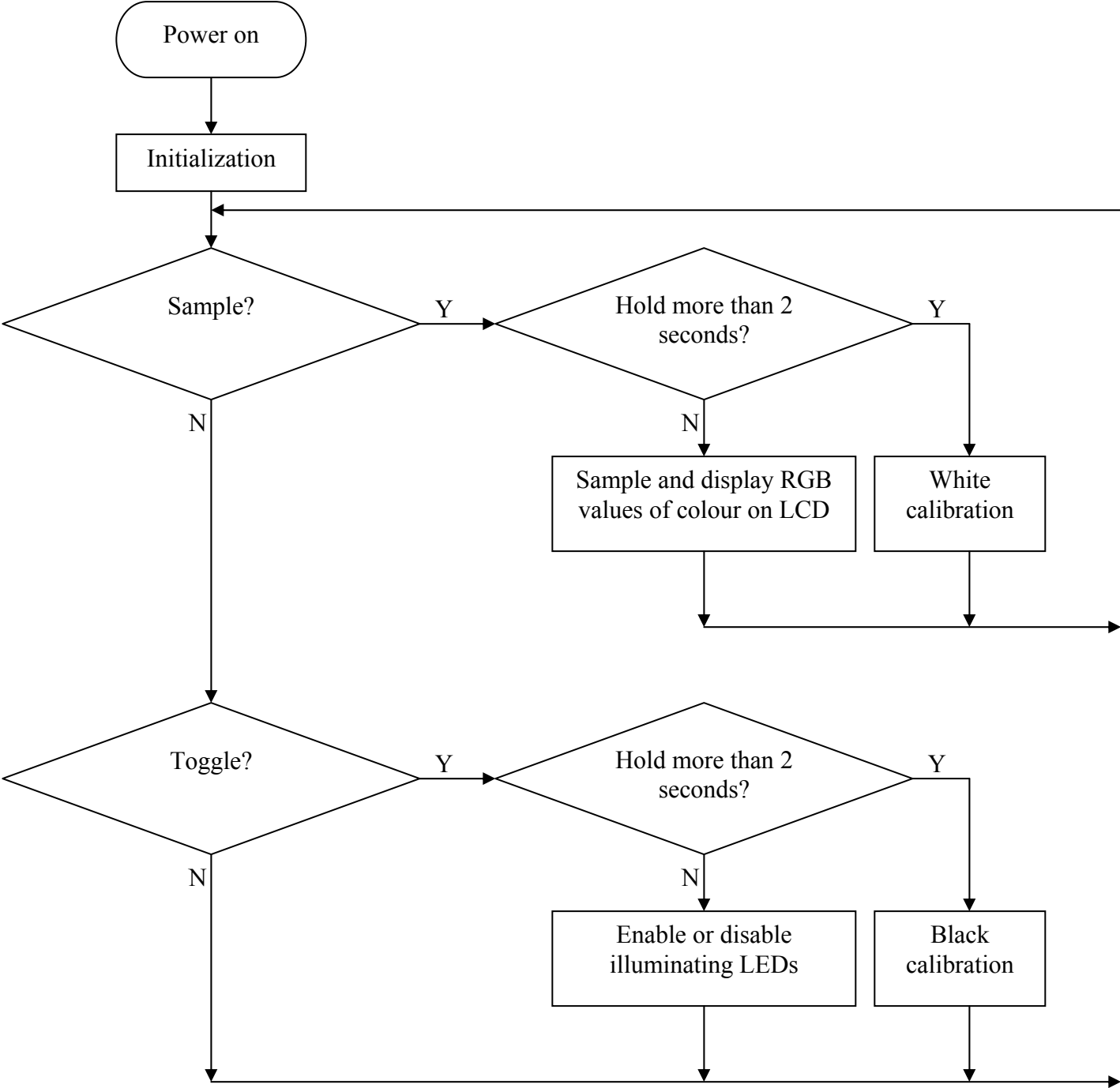
## **7. Conclusion**

This document has justified the careful decisions (and changes in decisions) ColourSense has made based on feasibility, practicality and usability. The technical difficulties encountered were documented and future improvements to the project were also discussed. Despite our struggles, we have succeeded in producing a functional prototype. ColourIt is highly precise; it can be used repeatedly to sample the colour of any area and it will produce very consistent results. ColourIt is compact and portable for the user; it fits into most pockets and purses and is powered by standard batteries. Finally, ColourIt is user-friendly as it offers the user many levels of options, depending on their applications.

## 8. References

- [1] Bridgeport – Quality Conduit and Cable Fittings, “Not All Conduit Bodies Can Contain Splices”  
<http://www.bptfittings.com/contractors/condbdy.shtml>, January 2005
- [2] GretagMacbeth – Color Checker Charts  
[http://www.gretagmacbeth.com/index/products/products\\_color-standards.htm](http://www.gretagmacbeth.com/index/products/products_color-standards.htm), 2004

## Appendix A: Software Flow Charts



## Appendix B: Instruction Set of PIC18F452

Mnemonic	Operands	Description	Words	Cycles	Status bits affected
movlw	k	Move literal value to WREG	1	1	-
movwf	f(, BANKED)	Move WREG to f	1	1	-
movff	fs, fb	Move fs to fb(both with full 12-bit addresses)	2	2	-
movf	f, F/W(, BANKED)	Move f to F or WREG	1	1	Z, N
lfsr	i, k	Load FSRi with full 12-bit address, where i_0 to 2	2	2	-
clrf	f(, BANKED)	Clear f	1	1	Z
setf	f(, BANKED)	0xff → f	1	1	-
movlb	k	Move literal value to BSR_3 : 0_, where k_0 to 15, to set the bank for direct addressing	1	1	-
swapf	f, F/W(, BANKED)	Swap nibbles of f, putting result in F or WREG	1	1	-
bcf	f, b(, BANKED)	Clear bit b of register f, where b_0 to 7	1	1	-
bsf	f, b(, BANKED)	Set bit b of register f, where b_0 to 7	1	1	-
btg	f, b(, BANKED)	Toggle bit b of register f, where b_0 to 7	1	1	-
rlcf	f, F/W(, BANKED)	Copy f into F or WREG; rotate F or WREG left through carry bit (9-bit rotate left)	1	1	C, N, Z
rlncf	f, F/W(, BANKED)	Copy f into F or WREG; rotate F or WREG left without carry bit (8-bit rotate left)	1	1	N, Z
rrcf	f, F/W(, BANKED)	Copy f into F or WREG; rotate F or WREG right through carry bit (9-bit rotate right)	1	1	C, N, Z
rrncf	f, F/W(, BANKED)	Copy f into F or WREG; rotate F or WREG right without carry bit (8-bit rotate right)	1	1	N, Z
incf	f, F/W(, BANKED)	Increment f, putting result in F or WREG	1	1	C, DC, Z, OV, N
decf	f, F/W(, BANKED)	Decrement f, putting result in F or WREG	1	1	C, DC, Z, OV, N
comf	f, F/W(, BANKED)	Complement f, putting result in F or WREG	1	1	Z, N
negf	f(, BANKED)	Change sign of a twos-complement-coded number	1	1	C, DC, Z, OV, N
andlw	k	AND literal value into WREG	1	1	Z, N
andwf	f, F/W(, BANKED)	AND WREG with f, putting result in F or WREG	1	1	Z, N
iorlw	k	Inclusive-OR literal value into WREG	1	1	Z, N
iorwf	f, F/W(, BANKED)	Inclusive-OR WREG with f, putting result in F or WREG	1	1	Z, N
xorlw	k	Exclusive-OR literal value into WREG	1	1	Z, N
xorwf	f, F/W(, BANKED)	Exclusive-OR WREG with f, putting result in F or WREG	1	1	Z, N
addlw	k	Add literal value into WREG	1	1	C, DC, Z, OV, N
addwf	f, F/W(, BANKED)	Add WREG and f, putting result in F or WREG	1	1	C, DC, Z, OV, N
addwfc	f, F/W(, BANKED)	Add WREG and f and carry bit, putting result in F or WREG	1	1	C, DC, Z, OV, N
daw		Decimal adjust sum of two packed BCD digits to correct packed BCD result in WREG	1	1	C
sublw	k	Subtract WREG from literal value, putting result in WREG	1	1	C, DC, Z, OV, N
subwf	f, F/W(, BANKED)	Subtract WREG from f, putting result in f or WREG	1	1	C, DC, Z, OV, N
subwfb	f, F/W(, BANKED)	Subtract WREG and borrow bit from f, putting result in F or WREG	1	1	C, DC, Z, OV, N
subfwb	f, F/W(, BANKED)	Subtract f and borrow bit from WREG, putting result in F or WREG	1	1	C, DC, Z, OV, N
mullw	k	Multiply WREG with literal value, putting result in PRODH : PRODL (WREG remains unchanged)	1	1	-
mulwf	f(, BANKED)	Multiply WREG with f, putting result in PRODH : PRODL (WREG and f remain unchanged)	1	1	-



Mnemonic	Operands	Description	Words	Cycles	Status bits affected
btfscc	f, b(, BANKED)	Test bit b of register f, where b_0 to 7; skip if clear	1	1/2	-
btfss	f, b(, BANKED)	Test bit b of register f, where b_0 to 7; skip if set	1	1/2	-
bra	label	Branch to labeled instruction (within _64 one-word instructions)	1	2	-
goto	label	Go to labeled instruction (anywhere)	2	2	-
bc	label	If carry (C_1), then branch to labeled instruction (within _64 one-word instructions)	1	1/2	-
bnc	label	If no carry (C_0), then branch to labeled instruction (within _64 one-word instructions)	1	1/2	-
bz	label	If zero (Z_1), then branch to labeled instruction (within _64 one-word instructions)	1	1/2	-
bnz	label	If not zero (Z_0), then branch to labeled instruction (within _64 one-word instructions)	1	1/2	-
bn	label	If negative (N_1), then branch to labeled instruction (within _64 one-word instructions)	1	1/2	-
bnn	label	If not negative (N_0), then branch to labeled instruction (within _64 one-word instructions)	1	1/2	-
bov	label	If overflow (OV_1), then branch to labeled instruction (within _64 one-word instructions)	1	1/2	-
bnov	label	If no overflow (OV_0), then branch to labeled instruction (within _64 one-word instructions)	1	1/2	-
cpfseq	f(, BANKED)	Skip if f is equal to WREG	1	1/2	-
cpfsgt	f(, BANKED)	Skip if f is greater than WREG (unsigned compare)	1	1/2	-
cpflt	f(, BANKED)	Skip if f is less than WREG (unsigned compare)	1	1/2	-
tstfsz	f(, BANKED)	Test f; skip if zero	1	1/2	-
decfsz	f, F/W(, BANKED)	Decrement f, putting result in F or WREG; skip if zero	1	1/2	-
defsnz	f, F/W(, BANKED)	Decrement f, putting result in F or WREG; skip if not zero	1	1/2	-
incfsz	f, F/W(, BANKED)	Increment f, putting result in F or WREG; skip if zero	1	1/2	-
infsnz	f, F/W(, BANKED)	Increment f, putting result in F or WREG; skip if not zero	1	1/2	-
rcall	label	Call labeled subroutine (within _512 one-word instructions)	1	2	-
call	label	Call labeled subroutine (anywhere)	2	2	-
call	label, FAST	Call labeled subroutine (anywhere); copy state to shadow registers: (WREG)→WS, (STATUS)→STATUSS, (BSR)→BSRS	2	2	-
return		Return from subroutine	1	2	-
return	FAST	Return from subroutine; restore state from shadow registers: (WS)→WREG, (STATUSS)→STATUS, (BSRS)→BSR	1	2	C, DC, Z, OV, N
retlw	k	Return from subroutine, putting literal value in WREG	1	2	-
retfie		Return from interrupt; reenale interrupts	1	2	-
retfie	FAST	Return from interrupt; restore state from shadow registers: (WS)→WREG, (STATUSS)→STATUS, (BSRS)→BSR; reenale interrupts	1	2	C, DC, Z, OV, N
push		Push address of next instruction onto stack	1	1	-
pop		Discard address on top of stack	1	1	-
clrwdt		Clear watchdog timer	1	1	-
sleep		Go into standby mode	1	1	-
reset		Software reset to same state as is achieved with the MCLR input	1	1	C, DC, Z, OV, N
nop		No operation	1	1	-
tblrd*		Read from program memory location pointed to by TBLPTR into TABLAT	1	2	-
tblrd*_		Read from program memory location pointed to by TBLPTR into TABLAT, then increment TBLPTR	1	2	-

tblrd*_	Read from program memory location pointed to by TBLPTR into TABLAT, then decrement TBLPTR	1	2	-
tblrd*_	Increment TBLPTR, then read from program memory location pointed to by TBLPTR into TABLAT	1	2	-