



## BÁO CÁO ĐỒ ÁN VDK - báo cáo đồ án

thương mại điện tử (Trường Cao đẳng Thực hành FPT)



Scan to open on Studocu

**TRƯỜNG ĐẠI HỌC BÁCH KHOA – ĐẠI HỌC ĐÀ NẴNG**  
**KHOA CƠ KHÍ**

-----



**BÁO CÁO**  
**ĐỒ ÁN VI ĐIỀU KHIỂN**

**ĐỀ TÀI: Thiết kế hệ thống phân loại sản phẩm theo màu sắc**

GV HƯỚNG DẪN: Ths. Trần Quang Khải

SV THỰC HIỆN : Cao Văn Đông

Trần Bá Hoàng

Lớp : 17CDT1

Nhóm : 17NH06B

ĐÀ NẴNG, 12-2020

## TÓM TẮT ĐỒ ÁN

Hệ thống phân loại sản phẩm ra đời trong quá trình phát triển qui trình sản xuất của các nhà máy xí nghiệp. Hệ thống có thể phân loại sản phẩm dựa vào các đặc tính như phân loại sản phẩm không đạt chất lượng, theo màu sắc, khối lượng, hình dạng...để có thể sắp xếp các sản phẩm có cùng đặc tính lại với nhau. Hệ thống này giúp cho sản xuất trở nên linh hoạt hơn, tiết kiệm thời gian và nguồn nhân lực. Việc nghiên cứu và cải thiện hiệu suất, tính ổn định và độ chính xác của hệ thống đang là vấn đề được quan tâm.

Đồ án này, chúng em tập trung vào việc thiết kế và chế tạo một mô hình phân loại sản phẩm theo màu sắc. Đồ án được chia làm 2 phần chính là phần thiết kế cơ khí và phần điều khiển tự động của mô hình. Phần Thiết kế cơ khí là một hệ thống băng tải dùng để di chuyển sản phẩm và phân cơ cấu chấp hành có nhiệm vụ thực hiện phân loại phẩm khi có tín hiệu từ hệ thống điều khiển. Phần điều khiển có nhiệm vụ nhận biết, phân tích, hiện thị sản phẩm và ra lệnh để điều khiển cơ cấu chấp hành.

Đề tài này nhóm em sử dụng vi điều khiển PIC 16F877A, cảm biến màu TCS3200, các cảm biến hồng ngoại, các servo và động cơ giảm tốc...dùng để điều khiển mô hình phân loại sản phẩm với hiệu suất nhỏ và giá cả phù hợp.

## LỜI CẢM ƠN

Qua một khoảng thời gian nghiên cứu và thực hiện, đến nay nhóm em đã hoàn thành xong đồ án môn học vi điều khiển với đề tài: “ *Hệ thống phân loại sản phẩm theo màu sắc* ” do giảng viên **Ths. Trần Quang Khải** hướng dẫn. Trong suốt quá trình nghiên cứu và thực hiện đề tài, nhóm em đã gặp rất nhiều khó khăn, thử thách và đã nhận được nhiều sự giúp đỡ và quan tâm nhiệt tình của Thầy.

Để hoàn thành được đồ án vi điều khiển cho phép nhóm em gửi lời cảm ơn đến các Thầy trong khoa Cơ Khí đã giảng dạy và truyền thụ những kiến thức quý báu, bổ ích thông qua các môn học ở trường. Giúp chúng em có một kiến thức cơ bản và một cách nhìn tổng quát hơn để hoàn thành tốt đồ án này.

Nhóm em xin chân thành cảm ơn đến Thầy **Ths. Trần Quang Khải** đã luôn luôn tận tình giúp đỡ, chỉ bảo và tạo mọi điều kiện cho nhóm trong suốt quá trình thực hiện đồ án.

Đà Nẵng, Ngày 12 tháng 12 năm 2020

Nhóm sinh viên thực hiện

Trần Bá Hoàng

Cao Văn Đông

## Mục lục

TÓM TẮT ĐỒ ÁN .....	1
LỜI CẢM ƠN.....	2
Mục lục .....	3
Chương 1: Giới thiệu chung về đề tài .....	4
1.1. Lý do chọn đề tài.....	4
1.2. Ứng dụng của đề tài.....	4
1.3. Hướng thực hiện đề tài.....	5
Chương 2: Giới thiệu về vi điều khiển và các loại cảm biến sử dụng.....	6
2.1. Vi điều khiển.....	6
2.2. Các loại cảm biến sử dụng.....	8
2.2.1. Cảm biến màu TCS3200.....	8
2.2.2. Cảm biến hồng ngoại E18 – D80NK.....	10
2.2.3. Động cơ servo MG90S .....	11
Chương 3 :Hoạt động của mô hình đồ án.....	12
3.1. Nguyên lí hoạt động của mô hình.....	12
3.2. Sơ đồ thuật toán.....	12
3.3. Kết quả thực tế.....	17
Chương 4: Kết luận.....	17
4.1. Các kiến thức đã ứng dụng.....	17
4.2. Phương án cải thiện.....	17
Phụ lục: Code của toàn bộ chương trình.....	18
Tài liệu tham Khảo.....	26

## Chương 1: Giới thiệu chung về đề tài

### 1.1. Lý do chọn đề tài

Trong bối cảnh đất nước ngày càng phát triển trong nhiều lĩnh vực, đặc biệt là trong sản xuất. Khi đất nước đang tiến hành đổi mới chúng ta từng bước tiếp cận với những công nghệ hiện đại, trong đó có điều khiển và tự động hóa. Bên cạnh đó, sự phát triển của ngành sản xuất ngày càng lớn và ngày càng có nhiều nhà máy xí nghiệp được xây lên làm cho việc cạnh tranh càng trở nên gay gắt hơn. Để đáp ứng cho nhu cầu của con người, các sản phẩm được sản xuất đòi hỏi phải đảm bảo chất lượng nhưng giá cả phải phù hợp. Yêu cầu các nhà máy phải luôn luôn cải cách quy trình sản xuất để tạo ra sản phẩm phù hợp với nhu cầu của thị trường.

Những công việc trong sản xuất thường có tính tuần hoàn và đòi hỏi phải tập trung cao độ nên dẫn đến công nhân hay bị mệt mỏi, chán nản và không thể tránh khỏi những sai sót.

Nhằm phục vụ cho nhiệm vụ hiện đại hóa quy trình sản xuất, hệ thống phân loại sản phẩm ra đời. Đây là một công cụ giúp thay thế cho con người giải quyết các vấn đề cơ bản trong sản xuất. Tạo ra một hệ thống phân loại sản phẩm hoàn chỉnh hơn, có độ tin cậy cao, hoạt động liên tục, giảm thời gian trì hoãn hệ thống và giảm được chi phí nhân công, tăng năng suất, đem lại lợi ích kinh tế cao và hiệu quả.

Ngoài việc giải quyết các vấn đề trong quy trình sản xuất, đồ án là một cơ hội để chúng em có thể củng cố lại kiến thức lý thuyết đã học trên trường và thấy được khả năng ứng dụng trong thực tiễn.

Vì vậy, nhóm em đã chọn đề tài: “ *Hệ thống phân loại sản phẩm theo màu sắc* ” để nghiên cứu và thực hiện nhằm giải quyết một vấn đề nhỏ trong sản xuất, giúp cải thiện chất lượng sản phẩm.

### 1.2. Ứng dụng của đề tài

Hệ thống phân loại sản phẩm theo màu sắc được ứng dụng rộng rãi trong cuộc sống hiện nay, đặc biệt trong các nhà máy, phân xưởng.

Hệ thống có thể phân loại các sản phẩm như : chai nhựa, bánh kẹo, các đồ gia dụng...trong các nhà máy vừa và nhỏ, các phân xưởng.

### 1.3. Hướng thực hiện đề tài

Để tiến hành nghiên cứu và thực hiện đề tài : “ *Thiết kế hệ thống phân loại sản phẩm theo màu sắc* ” cần làm những công việc sau:

- Tìm hiểu về nguyên lý hoạt động và cấu tạo của một quy trình phân loại sản phẩm.
- Lập sơ đồ khối về mô hình phân loại sản phẩm theo màu sắc

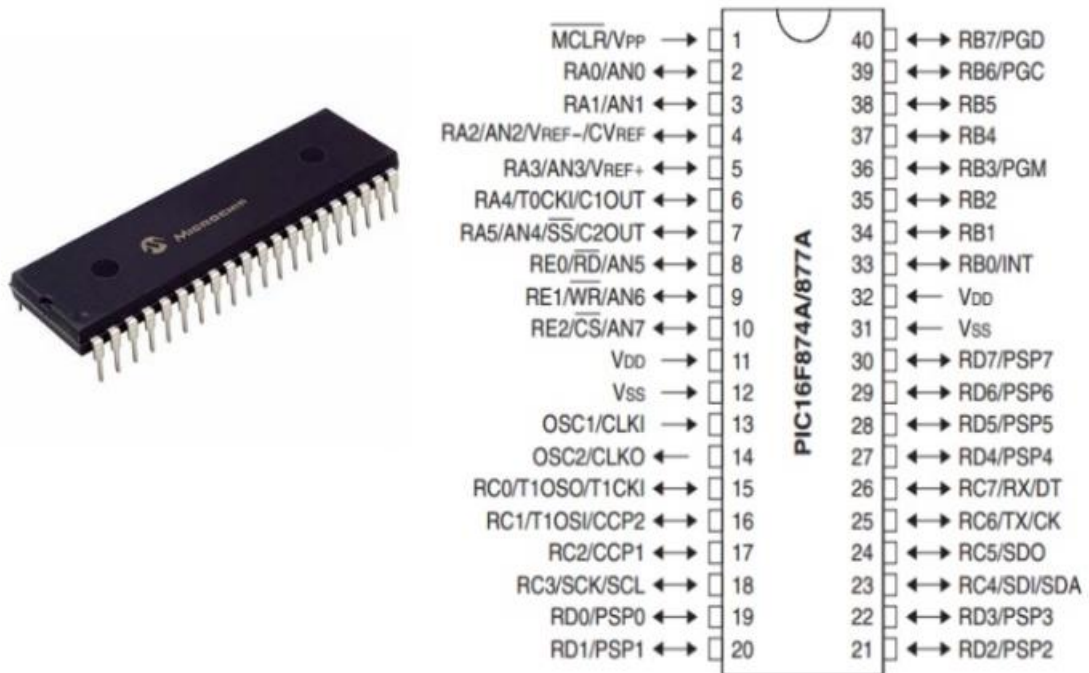
- Lập danh sách các cảm biến, linh kiện điện tử, vi điều khiển, các bộ phận cơ khí... cần áp dụng vào trong đề tài.
- Tìm hiểu về nguyên lý hoạt động, thông số kỹ thuật của các linh kiện được sử dụng.
- Tiến hành vẽ sơ đồ nguyên lý hoạt động của mạch điều khiển trên protus và mô hình mô phỏng nguyên lý hoạt động của hệ thống.
- Bắt đầu in mạch, lắp linh kiện và kiểm tra mạch điều khiển. Bên cạnh đó cũng tiến hành làm mô hình thực tế của một hệ thống phân loại sản phẩm theo màu sắc.
- Lập sơ đồ thuật toán hoạt động để tiến hành lập trình điều khiển mô hình.
- Cuối cùng tiến hành kiểm tra, thử nghiệm và sửa lỗi để hoàn thành được đề tài nghiên cứu.

## Chương 2: Giới thiệu về vi điều khiển và các loại cảm biến sử dụng.

### 2.1. Vi điều khiển

a. Tên: PIC 16F877A

b. Thông số cơ bản:



PIC 16F877A là dòng PIC khá phổ biến, khá đầy đủ tính năng phục vụ cho hầu hết tất cả các ứng dụng thực tế. Đây là dòng PIC khá dễ cho người mới làm quen với PIC có thể học tập và tạo nền tảng về hệ vi điều khiển PIC của mình..

Cấu trúc tổng quát của PIC 16F877A như sau:

- 8K Flash Rom.
- 368 bytes Ram.
- 256 bytes RFPROM.
- 5 port vào ra với tín hiệu điều khiển độc lập.
- 2 bộ định thời Timer0 và Timer2 8bit.
- 1 bộ định thời Timer1 16bit có thể hoạt động ở cả chế độ tiết kiệm năng lượng với nguồn xung clock ngoài.
- 2 bộ Capture/Compare/PWM.
- 1 bộ biến đổi Analog -> Digital 10 bit, 8 ngõ vào.
- 2 bộ so sánh tương tự.



- 1 vỏ định thời giám sát (Watch Dog Timer).
- 1 cổng song song 8 bit với các tín hiệu điều khiển.
- 1 cổng nối tiếp .
- 15 nguồn ngắt.

#### c. Ưu nhược điểm của PIC 16F877A

- Ưu điểm:
  - Được phổ biến rộng rãi.
  - Tích hợp sẵn nhiều bộ giao tiếp ngoại vi ADC, PWM,...
  - Sử dụng được nhiều loại ngôn ngữ để lập trình cho PIC như ngôn ngữ C, Assembly,...

#### d. Ngôn ngữ lập trình cho PIC và cách thức nạp chương trình cho PIC.

- Ngôn ngữ lập trình cho PIC rất đa dạng, có hai loại:
  - Ngôn ngữ lập trình cấp thấp – hợp ngữ: MPLAB ( được cung cấp miễn phí bởi nhà sản xuất Microchip).
  - Ngôn ngữ lập trình bậc cao: có nhiều loại, được phát triển trên ngôn ngữ C: CCS, HTPIC, PICBasic,...

Ưu điểm của hợp ngữ là giúp người học và lập trình hiểu rõ hơn về cấu trúc bên trong của vi điều khiển PIC, cũng như khả năng tối ưu hóa bộ nhớ chương trình. Tuy nhiên, tiếp cận hợp ngữ cả khả năng phát triển là hạn chế, mất thời gian. Do đó, trong nội dung của đồ án em sử dụng ngôn ngữ lập trình bậc cao CCS để sử dụng và phát triển.

Ưu điểm CCS:

- Kế thừa tất cả đặc điểm của ngôn ngữ C – là ngôn ngữ cơ bản quen thuộc mà sinh viên đã được học.
- Xây dựng sẵn các hàm phục vụ cho việc sử dụng dễ dàng các khối chức năng đặc biệt của vi điều khiển PIC như khối ADC, PWM, RS232, SPI.
- Có khả năng kết hợp với ngôn ngữ hợp ngữ, tạo sự mềm dẻo trong phát triển ứng dụng.
- Khả năng phát triển, nâng cấp ứng dụng là dễ dàng.
- Ngày càng được cập nhật với nhiều tính năng ưu việt và hiệu quả hơn.
- Mạch nạp cho PIC
  - Đây cũng là một dòng sản phẩm rất đa dạng cho vi điều khiển PIC. Có thể sử dụng các mạch nạp được cung cấp bởi nhà sản xuất là hãng Microchip như: PICSTART plus, MPLAB ICD 2, MPLAB PM 3, PRO MATE II. Có thể dùng các sản phẩm này để nạp cho vi điều khiển khác thông qua chương trình MPLAB. Dòng sản phẩm chính thống này có ưu thế là nạp được cho

tất cả các vi điều khiển PIC, tuy nhiên giá thành rất cao và thường gặp rất nhiều khó khăn trong quá trình mua sản phẩm.

- Ngoài ra do tính năng cho phép nhiều chế độ nạp khác nhau, còn có rất nhiều mạch nạp được thiết kế dành cho vi điều khiển PIC. Có thể sơ lược một số mạch nạp cho PIC như PIC kit2, PICkit3, brune,...



## 2.2. Các loại cảm biến sử dụng

### 2.2.1. Cảm biến màu TCS3200

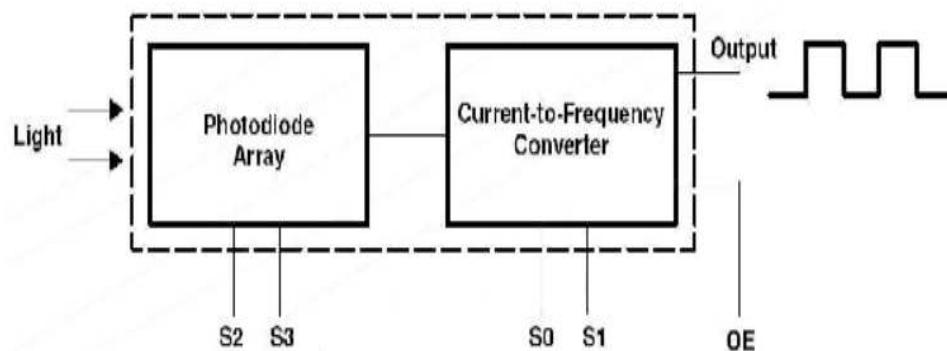


#### a. Thông số cơ bản

- Điện áp cung cấp từ 2,7V~ 5,5V.
- Chuyển đổi từ cường độ ánh sáng sang tần số với độ phân giải cao.
- Lập trình lựa chọn bộ lọc màu sắc khác nhau và dạng tần số xuất ra.
- Điện năng tiêu thụ thấp. Giao tiếp trực tiếp với vi điều khiển.

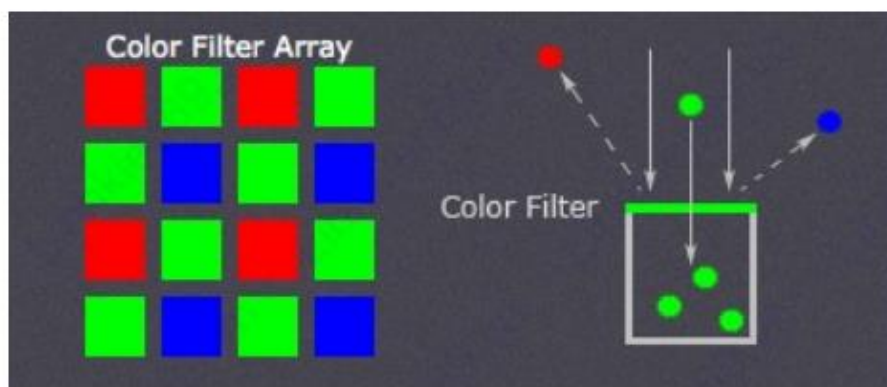
## b. Nguyên lý hoạt động

- Cấu tạo cảm biến TCS3200 gồm 2 khối như hình vẽ phía dưới:



- Khối đầu tiên là mảng ma trận 8x8 gồm các photodiode. Bao gồm 16 photodiode có thể lọc màu sắc xanh dương(Blue), 16 photodiode có thể lọc màu đỏ (Red), 16 photodiode màu xanh lá (Green) và 16 photodiode trắng không lọc (Clear). Tất cả photodiode cùng màu đều được kết nối song song nhau, và được đặt xen kẽ nhau nhằm mục đích chống nhiễu.

- Bản chất 4 loại photodiode trên như là các bộ lọc ánh sáng có màu sắc khác nhau. Có nghĩa nó chỉ tiếp nhận các ánh sáng có cùng màu với loại photodiode tương ứng và không tiếp nhận các ánh sáng có màu sắc khác.



- Tần số đầu ra của linh kiện điện tử TCS3200 trong khoảng 2Hz ~ 500KHz. Tần số đầu ra có dạng xung vuông với tần số khác nhau khi mà màu sắc khác nhau và cường độ ánh sáng là khác nhau.

## c. Cách nối dây

- S0,S1: Đầu vào chọn tỉ lệ tần số đầu ra.
- S2,S3: Đầu vào chọn kiểu photodiode.
- OE: Đầu vào cho phép xuất tần số ở chân OUT.

- OUT: Đầu ra là tần số thay đổi phụ thuộc vào cường độ và màu sắc.

#### **d. Ứng dụng**

- Phân biệt các màu khác nhau.
- Đối với đồ án này cảm biến màu TCS3200 dùng để phân biệt 3 màu đỏ, xanh dương, tím.

#### **2.2.2. Cảm biến hồng ngoại E18 – D80N**



##### **a. Thông số cơ bản:**

- Dạng đóng ngắt : Thường mở (NO – Normally Open)
- Số dây : 3 dây ( 2 dây nguồn và 1 dây tín hiệu)
- Nguồn điện cung cấp : 5VDC
- Khoảng cách phát hiện : 3 ~ 80cm
- Có thể điều chỉnh khoảng cách qua biến trở.
- Dòng kích ngõ ra.
- Ngõ ra cảm biến ở dạng cực thu hở NPN nên cần thêm trở kéo lên VCC (khoảng 1~10k Ohm) trước khi giao tiếp với Vi điều khiển.
- Chất liệu sản phẩm: nhựa.
- Có led hiển thị ngõ ra màu đỏ.
- Kích thước: 18 x 68mm

##### **b. Nguyên lý hoạt động:**

- Nguyên lý hoạt động đơn giản: mắt phát phát ra ánh sáng nếu gặp vật cản thì phản xạ lại mắt thu nhận được và trả lại tín hiệu về.

##### **c. Cách nối dây**

- Màu vàng: dây tín hiệu.
- Màu đỏ : VCC= 5V.

- Màu đen: GND.
- Khi có vật cảm biến sẽ trả về vi điều khiển mức thấp, khi không có sẽ trả về mức cao.

#### d. Ứng dụng

- Phát hiện các vật thể khác nhau.
- Đối với trong đồ án này cảm biến hồng ngoại được dùng để phát hiện phôi đi qua truyền tín hiệu về vi điều khiển để điều khiển các servo gạt phôi vào đúng vị trí.

### 2.3. Động cơ servo MG90S



#### a. Thông số kỹ thuật

- Model: MG90S servo
- Điện áp hoạt động: 4.8 ~ 6VDC.
- Mômen xoắn: 1.8kg/cm(4.8V), 2.2kg/cm(6V)
- Tốc độ: 0.1sec/60degree(4.8v), 0.08sec/60degree(6v)
- Bánh răng: Kim loại
- Độ dài dây nối: 175mm
- Trọng lượng: 13.4g
- Kích thước: 22.8 x 12.2 x 28.5mm

#### b. Nguyên lí hoạt động

- Động cơ servo MG90S có tích hợp sẵn Driver điều khiển động cơ bên trong nên có thể dễ dàng điều khiển góc quay bằng phương pháp điều chế độ rộng xung PWM.

#### c. Ứng dụng

- Phù hợp cho nhiều ứng dụng khác nhau: Robot cánh tay máy, robot nhện, cơ cấu chuyển hướng, cơ cấu quay góc,...

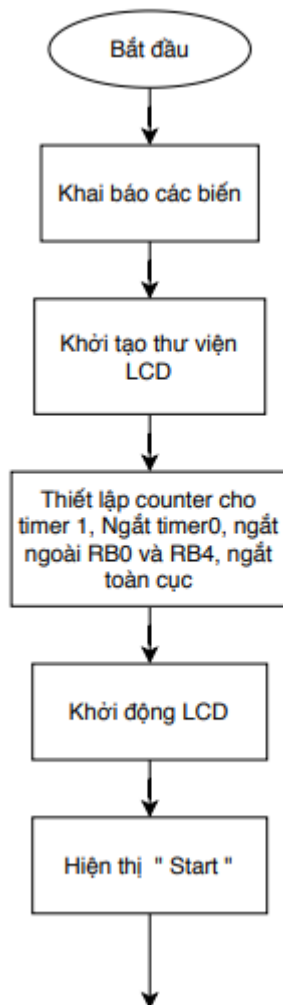
## Chương 3: Hoạt động của mô hình đồ án

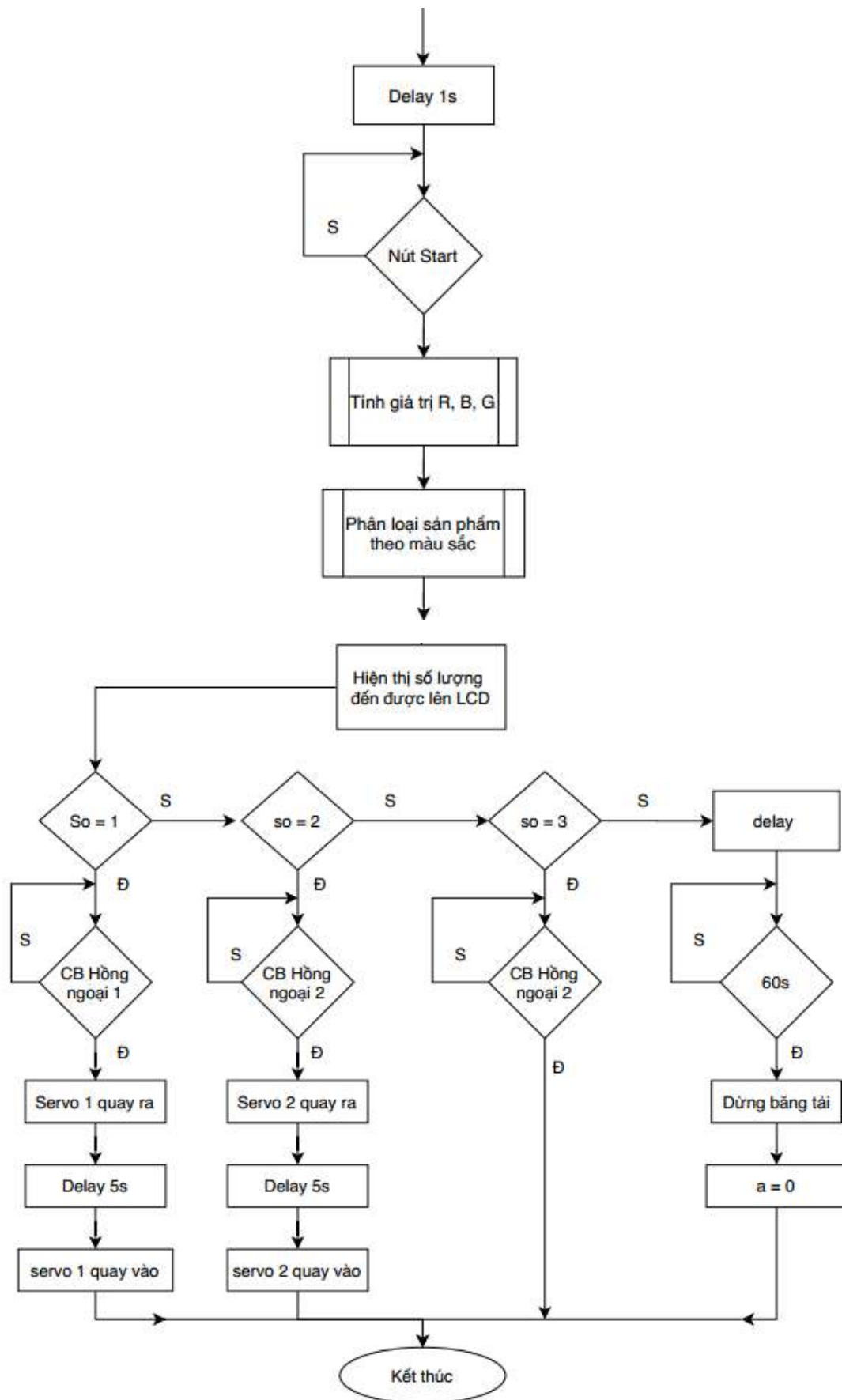
### 3.1. Nguyên lí hoạt động của mô hình

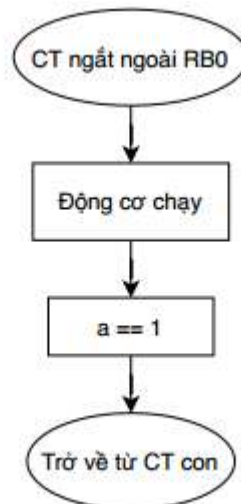
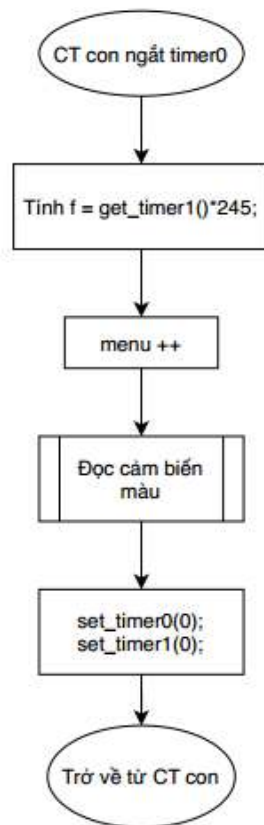
Hệ thống có thể đếm và phân loại các sản phẩm theo màu sắc. Khi nhấn nút Start thì băng tải bắt đầu hoạt động. Bây giờ sản phẩm được đưa vào băng tải và sau đó đi qua cảm biến màu TCS3200. Nhờ có cảm biến màu ta có thể phân loại màu sắc của các sản phẩm. Khi đó TCS3200 sẽ gửi tín hiệu đến con vi điều khiển PIC 16F877A và vi điều khiển sẽ gửi tín hiệu đến servo để đẩy sản phẩm xuống thùng với màu tương ứng. Đồng thời, vi điều khiển cũng gửi tín hiệu đến khối hiển thị LCD để hiển thị số lượng sản phẩm đếm được.

Nếu trong quá trình hoạt động xảy ra lỗi ta có thể nhấn nút Stop để dừng băng tải và tiến hành sửa chữa. Nếu trong một khoảng thời gian nhất định mà sản phẩm hết hay chưa cấp cho băng tải thì động cơ sẽ dừng và kết thúc quá trình làm việc.

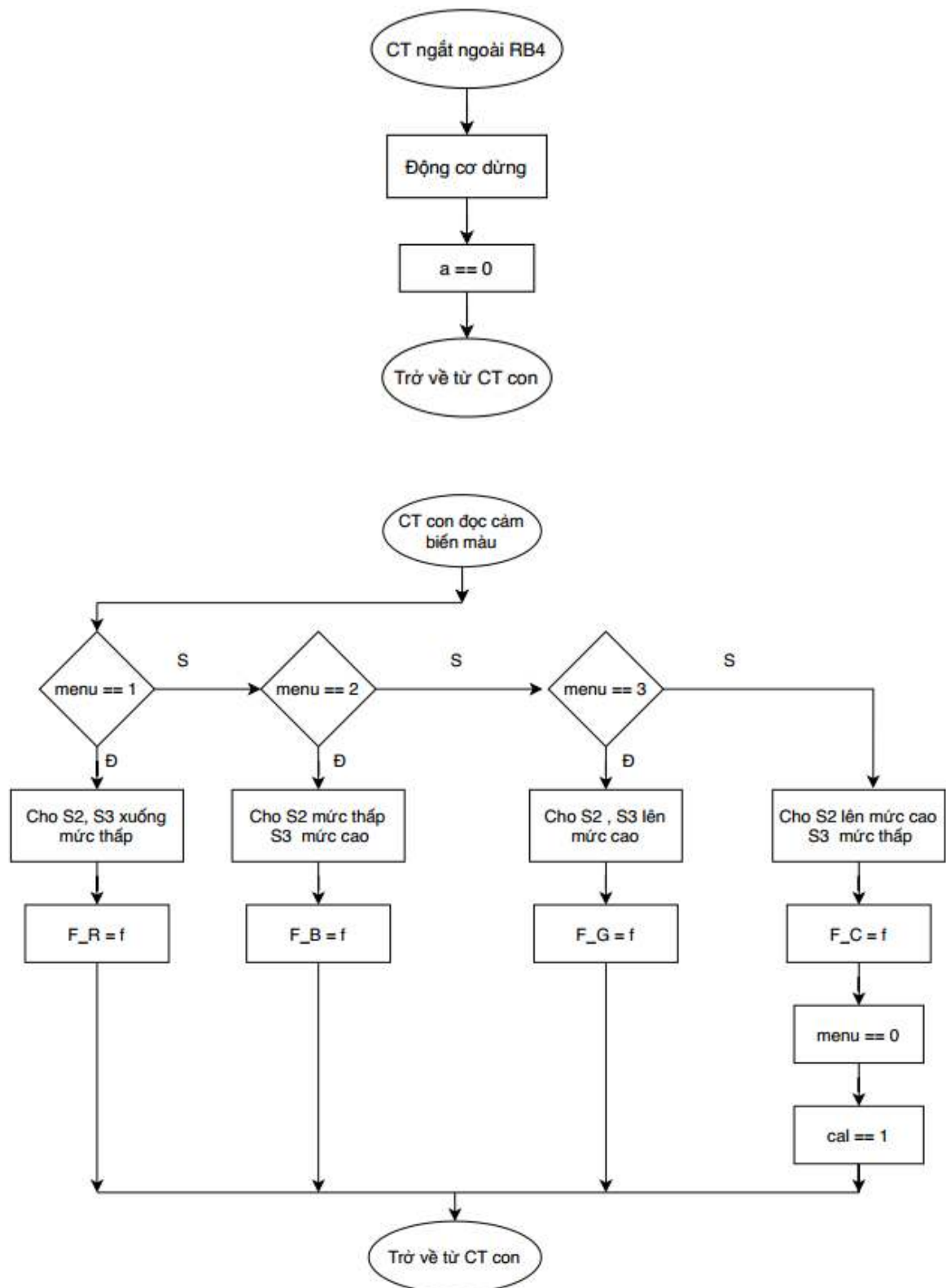
### 3.2. Sơ đồ thuật toán

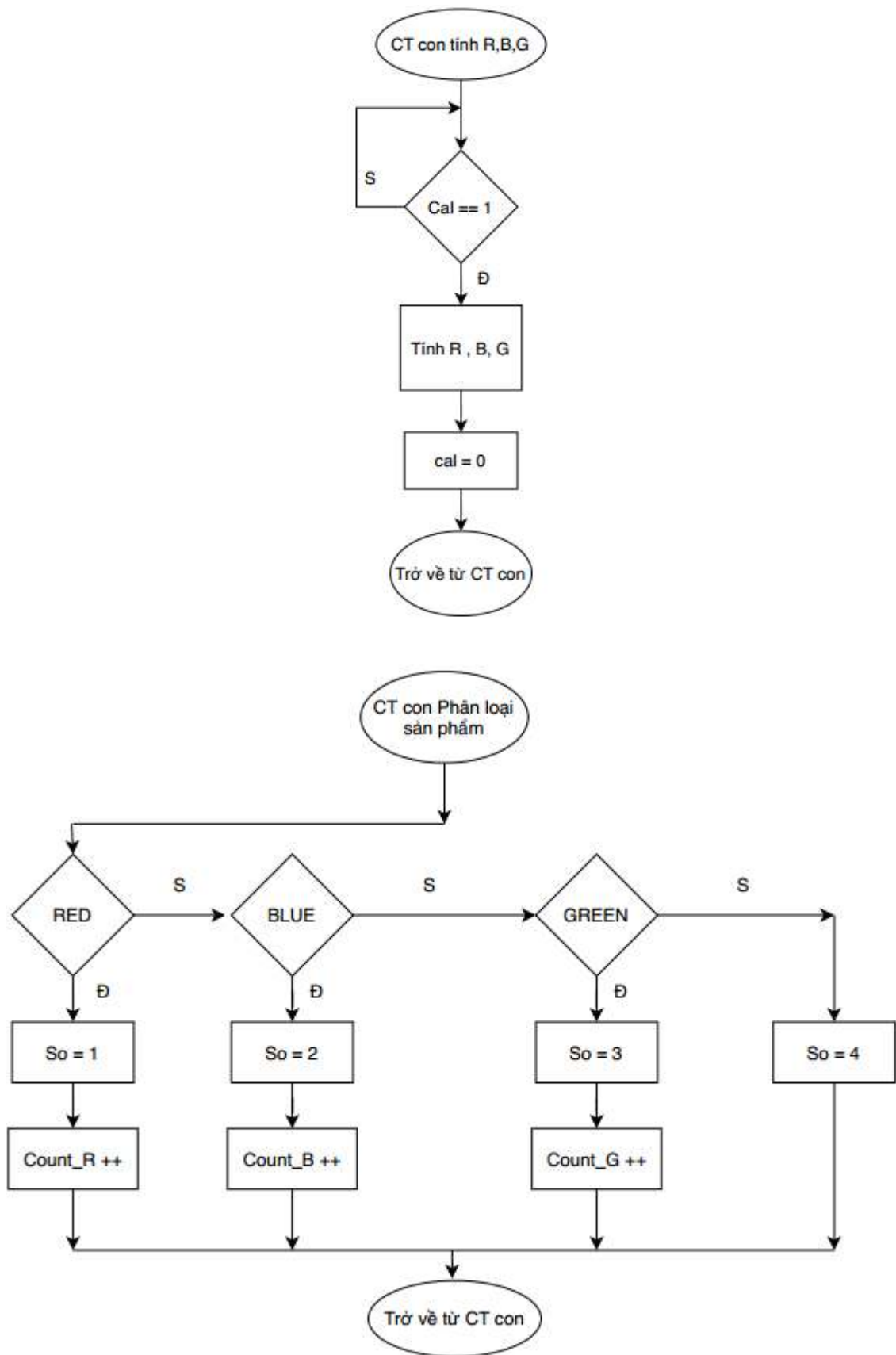












### 3.3. Kết quả thực tế

Phần thiết kế cơ khí và mạch điều khiển tạm thời đã hoàn thiện theo đúng theo yêu cầu.

Mô hình hoạt động có thể phân loại các sản phẩm theo màu sắc và hiện thị số lượng sản phẩm đếm được lên màn hình LCD.

Trong quá trình hoạt vẫn còn nhiều hạn chế và sai sót về khâu xử lí tín hiệu của cảm biến màu. Do bị nhiễu bởi nguồn sáng của môi trường xung quanh làm cho tần số thu được từ sản phẩm không ổn định.

## Chương 4: Kết luận

### 4.1 Những kiến thức đã áp dụng

Để hoàn thành đồ án vi điều khiển, nhóm em đã áp dụng những kiến thức đã học ở trên trường và học hỏi từ bên ngoài.

Lý thuyết cơ bản từ các môn như Thiết kế máy, truyền động cơ khí... và kiến thức vẽ mô phỏng trên phần mềm solidwords để thiết kế phần cơ khí.

Thông qua môn cảm biến công nghiệp giúp nhóm hiểu rõ hơn về các cảm biến để thiết kế mạch điều khiển.

Môn kỹ thuật vi điều khiển giúp nhóm biết cách lập trình điều khiển các cảm biến trên pic 16f877a.

### 4.2 Phương án cải thiện

Trong quá trình làm việc, hệ thống vẫn gặp nhiều sai sót và hạn chế trong khi xử lí tín hiệu do ảnh hưởng bởi ánh sáng xung quanh. Để khắc phục lỗi xử lí tín hiệu ta có thể thay vì dùng cảm biến màu TCS3200 để phân biệt màu sắc bằng xử lí ảnh.

Sử dụng cánh tay gấp hoặc xi lanh để thực hiện đưa sản phẩm đã phân loại vào thùng chứa thay vì dùng servo giúp cho năng suất cao hơn, hiệu quả hơn.

## Phụ lục

```
#include<16f877a.h>
#include<string.h>
#include<stdlib.h>
#Fuses HS, NOWDT, NOPROTECT, BROWNOUT, PUT, NOLVP
#use delay(crystal = 8MHz)
#use rs232(baud = 9600, parity =N, xmit = pin_c6, rcv = pin_c7, bits = 8)
#byte T1CON = 0x10

float f = 0, F_R =0, F_B =0, F_G=0, F_C=0;
int R=0,B =0,G=0;
int1 cal = 0;
char menu = 0, c;
int so, a = 0 ,str[2] ;
int count = 0, count_r = 0, count_b = 0, count_g = 0;

void Read_Color();
void Calculate_RBG();
void classify();
void Display_Lcd();
void Controll_Servo1_out();
void Controll_Servo1_in();
void Controll_Servo2_out();
void Controll_Servo2_in();

#define S2 PIN_C2
#define S3 PIN_C3
#define Servo2 PIN_A1
#define Servo1 PIN_B2
#define Sensor2 PIN_A0
#define Sensor1 PIN_B3

#define LCD_RS_PIN PIN_D6
#define LCD_RW_PIN PIN_D5
```

```
#define LCD_ENABLE_PIN  PIN_D4
#define LCD_DATA4       PIN_C5
#define LCD_DATA5       PIN_C4
#define LCD_DATA6       PIN_D3
#define LCD_DATA7       PIN_D2
#include<lcd.c>

#int_ext
void Ngat_ext(){
    output_high(pin_d0);
    a = 1;
}

#int_rb
void Ngat_rb4(){
    output_low(pin_d0);
    a = 0;
}

#int_timer0
void Ngat_timer0()
{
    f = get_timer1()*245;
    menu++;
    Read_Color();
    set_timer1(0);
    set_timer0(0);
}

void main()
{
    output_low(pin_d0); //motor_stop()
    setup_timer_0(T0_INTERNAL|T0_DIV_32);//4.08ms
    T1CON = 0X0B;
    set_timer0(0);
```

```
set_timer1(0);
output_low(S2);
output_low(S3);

set_tris_b(0x11);
ext_int_edge(h_to_l);
enable_interrupts(int_ext);
enable_interrupts(int_rb);
enable_interrupts(INT_timer0);
enable_interrupts(GLOBAL);

lcd_init();
lcd_putc('\f');
lcd_gotoxy(1,1);
lcd_putc("Starts");
delay_ms(1000);
lcd_putc('\f');
while(true)
{
    while(a == 1){
        Calculate_RGB();
        classify();
        Display_Lcd();
        if(so == 1){//red
            while(input(Sensor1)==1);
            Controll_Servo1_out();
            delay_ms(5000);
            Controll_Servo1_in();
            count = 0;
        }
        else if(so == 2){//blue
            while(input(Sensor2) == 1);
            Controll_Servo2_out();
            delay_ms(5000);
            Controll_Servo2_in();
```

```
        count = 0;
    }
    else if(so == 3){//green
        count = 0;
    }
    else{
        delay_ms(1);
        count ++ ;
        if(count == 60000){
            count = 0;
            output_low(pin_d0);//Motor_stop()
            a = 0;
        }
    }
}
}
}
}

void Read_Color()
{
    if(menu == 1)//red
    {
        output_low(S2);
        output_low(S3);
        F_R=f;
        output_low(S2);
        output_high(S3);
    }
    else if(menu == 2)//blue
    {
        output_low(S2);
        output_high(S3);
        F_B=f;
        output_high(S2);
        output_high(S3);
    }
}
```

```
    else if(menu ==3)//green
    {
        output_high(S2);
        output_high(S3);
        F_G=f;
        output_high(S2);
        output_low(S3);
    }
    else
    {
        output_high(S2);
        output_low(S3);
        F_C=f;
        menu=0;
        cal = 1;
        output_low(S2);
        output_low(S3);
    }
}

void Calculate_RBG()
{
    if(cal == 1)
    {
        R = (F_R*100)/F_C;
        B = (F_B*100)/F_C;
        G = (F_G*100)/F_C;
        cal = 0;
    }
}

void classify()
{
    if( R >= 50 && R <= 57 && B>=20 && B<=28 && G >= 14 && G <= 22) so = 1 ;
    else if ( R >= 12&&R <= 20 && B >= 49 && B<= 56 && G >= 23 && G<=27) so = 2;
```



```
else if ( R >= 25&& R<=33 && B >= 2 && B <= 35 && G >= 30 && G <= 44) so = 3;
else so = 4;
if(so == 1){//RED
    count_r++;
    if(count_r == 10) count_r = 0;
}else if(so ==2){ //Blue
    count_b++;
    if(count_b == 10) count_b = 0;
}else if(so == 3){ //Green
    count_g++;
    if(count_g == 10) count_g = 0;
}else{ //ERROR
}
delay_ms(100);
}
```

```
void Display_Lcd(){
    lcd_putc("\f");
    // Red
    lcd_gotoxy(1,1);
    lcd_putc("RED");
    sprintf(str,"%u",count_r);
    if(count_r<10)str[1]=' ';
    lcd_gotoxy(1,2);
    lcd_putc(str[0]);
    lcd_gotoxy(2,2);
    lcd_putc(str[1]);
    //Blue
    lcd_gotoxy(6,1);
    lcd_putc("BLUE");
    sprintf(str,"%u",count_b);
    if(count_b<10)str[1]=' ';
    lcd_gotoxy(6,2);
    lcd_putc(str[0]);
    lcd_gotoxy(7,2);
```

```
    lcd_putc(str[1]);  
    //Green  
    lcd_gotoxy(11,1);  
    lcd_putc("GREEN");  
    sprintf(str,"%u",count_g);  
    if(count_g<10)str[1]=' '  
    lcd_gotoxy(11,2);  
    lcd_putc(str[0]);  
    lcd_gotoxy(12,2);  
    lcd_putc(str[1]);  
}
```

```
void Controll_Servo1_out()
```

```
{  
  
    output_high(Servo1);  
    delay_us(2500);  
    output_low(Servo1);  
    delay_us(17500);  
  
}
```

```
void Controll_Servo1_in()
```

```
{  
  
    output_high(Servo1);  
    delay_us(1000);  
    output_low(Servo1);  
    delay_us(19000);  
  
}
```

```
void Controll_Servo2_out()
```

```
{  
    output_high(Servo2);
```

```
    delay_us(2500);  
    output_low(Servo2);  
    delay_us(17500);  
}  
void Controll_Servo2_in()  
{  
    output_high(Servo2);  
    delay_us(1000);  
    output_low(Servo2);  
    delay_us(19000);  
}
```

## **Tài liệu tham khảo**

- [1] Đặng Phước Vinh, Võ Như Thành, Giáo trình kỹ thuật vi điều khiển PIC, Nhà xuất bản xây dựng, 2019.
- [2] [alldatasheet.com](http://alldatasheet.com)
- [3] [howtomechatronics.com/tutorials/arduino/arduino-color-sensing-tutorial-tcs230-tcs3200-color-sensor/](http://howtomechatronics.com/tutorials/arduino/arduino-color-sensing-tutorial-tcs230-tcs3200-color-sensor/)
- [4] [www.arduino.cc](http://www.arduino.cc)