**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT**

**THÀNH PHỐ HỒ CHÍ MINH**

**BÁO CÁO ĐỒ ÁN**

**Môn học: 2**



Đề tài:

THIẾT KẾ VÀ THI CÔNG HỆ THỐNG ĐẾM VÀ PHÂN LOẠI SẢN PHẨM THEO CHIỀU CAO

GVHD : Th.S Nguyễn Đình Phú

SVTH : Nguyễn Văn Thiện - 19161292

Tp.Hồ Chí Minh, ngày 20 tháng 12 năm 2022

# LỜI CAM ĐOAN

Đề tài này là do tôi tự thực hiện dựa vào một số tài liệu và không sao chép từ tài liệu hay công trình đã có trước đó. Nếu có sao chép tôi hoàn toàn chịu trách nhiệm.

Sinh viên thực hiện đề tài

Nguyễn Văn Thiện

# LỜI CẢM ƠN

Trong thời gian làm đồ án môn học 2, em đã nhận được nhiều sự giúp đỡ, đóng góp ý kiến và chỉ bảo nhiệt tình của thầy cô, gia đình và bạn bè.

Em xin gửi lời cảm ơn chân thành đến Thầy Nguyễn Đình Phú- giảng viên hướng dẫn đã tận tình hướng dẫn, chỉ bảo em trong suốt quá trình làm đồ án.

Bên cạnh đó, em xin chân thành cảm ơn gia đình, bạn bè đã luôn quan tâm, giúp đỡ em trong suốt quá trình thực hiện.

Với điều kiện thời gian cũng như kinh nghiệm còn hạn chế của một sinh viên, đề tài này không thể tránh được những thiếu sót. Em rất mong nhận được sự chỉ bảo, đóng góp ý kiến của các thầy để em có điều kiện bổ sung, nâng cao ý thức của mình, phục vụ tốt hơn công tác thực tế sau này.

Sinh viên thực hiện đề tài

Nguyễn Văn Thiện

ĐIỂM SỐ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tiêu chí | Nội dung | Bố cục | Trình bày | Tổng |
|  |  |  |  |  |

NHẬN XÉT

………………………………………………………………………………………

………………………………………………………………………………………

………………………………………………………………………………………

………………………………………………………………………………………

………………………………………………………………………………………

………………………………………………………………………………………

………………………………………………………………………………………

………………………………………………………………………………………

………………………………………………………………………………………

………………………………………………………………………………………

………………………………………………………………………………………

………………………………………………………………………………………

*Tp. Hồ Chí Minh,* ngày tháng 12 năm 2022

GV HƯỚNG DẪN

(Ký và ghi rõ họ và tên)

MỤC LỤC

[LỜI CAM ĐOAN ……………………………………………………………………..2](#_Toc122373764)

[LỜI CẢM ƠN ……………………………………………………………………..3](#_Toc122373765)

[CHƯƠNG 1: TỔNG QUAN ……………………………………………………..11](#_Toc122373766)

[1. Lý do chọn đề tài …………………………………………………………….11](#_Toc122373767)

[2. Mục tiêu đề tài ……………………………………………………………………13](#_Toc122373768)

[3. Phương pháp nghiên cứu ……………………………………………………..13](#_Toc122373769)

[4. Giới hạn đề tài ……………………………………………………………………13](#_Toc122373770)

[5.Bố cục đề tài ……………………………………………………………………14](#_Toc122373771)

[CHƯƠNG 2: CƠ SỞ LÝ THUYẾT ……………………………………………...15](#_Toc122373772)

[2.1. Giới thiệu về hệ thống đếm và phân loại sản phẩm ………………………..15](#_Toc122373773)

[2.2 Tổng quan về cảm biến ……………………………………………………..15](#_Toc122373774)

[2.2.1 Giới thiệu chung về cảm biến ……………………………………………...15](#_Toc122373775)

[2.2.2 Phân loại các cảm biến ……………………………………………...16](#_Toc122373776)

[2.2.3 Vai trò của cảm biến ……………………………………………..............17](#_Toc122373777)

[2.3 Tổng quan về xử lý tín hiệu ……………………………………………...18](#_Toc122373778)

[2.3.1 Khái niệm xử lý tín hiệu ……………………………………………...18](#_Toc122373779)

[2.3.2 Phân loại xử lý tín hiệu ……………………………………………………..19](#_Toc122373780)

[2.4 Khái quát về vi điều khiển ……………………………………………..............20](#_Toc122373781)

[2.4.1 Giới thiệu về vi điều khiển ……………………………………………...20](#_Toc122373782)

[2.4.2 Các loại vi điều khiển và ứng dụng thực tiễn ………………………..21](#_Toc122373783)

[2.5 Tìm hiểu ngôn ngữ C ……………………………………………………..22](#_Toc122373784)

[2.5.1 Lịch sử hình thành ngôn ngữ C ………………………………………22](#_Toc122373785)

[2.5.2 Đặc điểm nổi bật và ứng dụng của ngôn ngữ C ………………………..23](#_Toc122373786)

[CHƯƠNG 3: THIẾT KẾ VÀ THI CÔNG HỆ THỐNG……………………………….26](#_Toc122373787)

[3.1 Giới thiệu hệ thống …………………………………………………………….26](#_Toc122373788)

3.1 Thiết kế sơ đồ khối ..............................................................................................26

[3.2.2. Chức năng của từng khối ……………………………………………...27](#_Toc122373790)

[3.3.1 Khối xử lý trung tâm ……………………………………………..............29](#_Toc122373792)

[3.3.2 Khối tín hiệu …………………………………………………………….34](#_Toc122373793)

[3.3.3 Khối phân loại …………………………………………………………….37](#_Toc122373794)

[3.3.4 Khối hiển thị …………………………………………………………….41](#_Toc122373795)

[3.3.5 Khối băng tải …………………………………………………………….44](#_Toc122373796)

[3.3.6 Tính toán, thiết kế mạch cho từng khối ……………………………….45](#_Toc122373797)

[ Khối nguồn ……………………………………………………………………45](#_Toc122373798)

[ Khối tín hiệu ……………………………………………………………………47](#_Toc122373799)

[ Khối xử lý trung tâm ……………………………………………………..48](#_Toc122373800)

[ Khối băng tải ……………………………………………………………………49](#_Toc122373801)

[ Khối phân loại …………………………………………………………….49](#_Toc122373802)

[ Khối hiển thị ……………………………………………………………………50](#_Toc122373803)

[3.3.7. Sơ đồ nguyên lý toàn mạch ……………………………………………...51](#_Toc122373804)

[3.3.8. Nguyên lý hoạt động ……………………………………………………..51](#_Toc122373805)

[3.4 Thi công hệ thống …………………………………………………………….52](#_Toc122373806)

[3.4.1 Thi công mạch …………………………………………………………….52](#_Toc122373807)

[3.4.2 Mạch in PCB …………………………………………………………….53](#_Toc122373808)

[3.5 Thi công mạch điều khiển ……………………………………………………..54](#_Toc122373809)

[3.5.1 Phần mềm lập trình Ardunio ……………………………………………...54](#_Toc122373810)

[3.5.2 Thi công mạch …………………………………………………………….59](#_Toc122373811)

[3.6. Lưu đồ giải thuật hệ thống ……………………………………………..............60](#_Toc122373812)

[3.6.1 Lưu đồ giải thuật …………………………………………………………….60](#_Toc122373813)

[3.7 Chương trình điều khiển ……………………………………………………..61](#_Toc122373814)

[CHƯƠNG 4: KẾT QUẢ\_KẾT LUẬN\_ĐÁNH GIÁ ……………………………….64](#_Toc122373815)

[4.1. Kết quả ……………………………………………………………………64](#_Toc122373816)

[4.1.1. Kết quả thực hiện ……………………………………………………..64](#_Toc122373817)

[4.1.2. Sản phẩm thực tế ……………………………………………………..65](#_Toc122373818)

[4.2 Kết luận ……………………………………………………………………66](#_Toc122373819)

[4.3 Hướng phát triển của đề tài ……………………………………………………..66](#_Toc122373820)

[TÀI LIỆU THAM KHẢO……………………………………………………………...67](#_Toc122373821)

DANH SÁCH BẢNG

[Bảng 3. 1 Danh sách các linh kiện 28](#_Toc122376811)

[Bảng 3. 2 Thông số của Ardunio Uno 32](#_Toc122376812)

[Bảng 3. 3 Thông số kĩ thuật của cảm biến hồng ngoại 36](#_Toc122376813)

[Bảng 3. 4 Thông số của động cơ servo 39](#_Toc122376814)

[Bảng 3. 5 Thông số của động cơ DC 41](#_Toc122376815)

[Bảng 3. 6 Chức năng chân của LCD 43](#_Toc122376816)

DANH SÁCH HÌNH ẢNH

[Hình 1. 1 Hệ thống băng tải vận chuyển đá những năm 90 12](#_Toc122378027)

[Hình 2. 1 Một số loại mạch cảm biến 16](#_Toc122378035)

[Hình 2. 2 Cảm biến hồng ngoại 17](#_Toc122378036)

[Hình 2. 3 Ứng dụng của cảm biến 18](#_Toc122378037)

[Hình 2. 4 Xử lý tín hiệu 18](#_Toc122378038)

[Hình 2. 5 Tín hiệu tương tự 19](#_Toc122378039)

[Hình 2. 6 Tín hiệu số 20](#_Toc122378040)

[Hình 2. 7 Vi điều khiển 21](#_Toc122378041)

[Hình 2. 8 Vi điều khiển dùng trong Ardunio 22](#_Toc122378042)

[Hình 2. 9 Ngôn ngữ lập trình C 23](#_Toc122378043)

[Hình 3. 1 Sơ đồ khối mô hình phân loại sản phẩm 27](#_Toc122378044)

[Hình 3. 2 Sơ đồ nguyên lý của hệ thống 28](#_Toc122378045)

[Hình 3. 3 Các phiên bản Ardunio 30](#_Toc122378046)

[Hình 3. 4 Ardunio Uno R3 31](#_Toc122378047)

[Hình 3. 5 Sơ đồ chân Ardunio Uno R3 33](#_Toc122378048)

[Hình 3. 6 Giao diện lập trình Ardunio IDE 34](#_Toc122378049)

[Hình 3. 7 Cảm biến hồng ngoại 35](#_Toc122378050)

[Hình 3. 8 Sơ đồ chân của cảm biến 36](#_Toc122378051)

[Hình 3. 9 Nguyên lý hoạt động của cảm biến 37](#_Toc122378052)

[Hình 3. 10 Động cơ servo 38](#_Toc122378053)

[Hình 3. 11 Hoạt động của động cơ servo 39](#_Toc122378054)

[Hình 3. 12 Động cơ DC 40](#_Toc122378055)

[Hình 3. 13 LCD 16 x 2 41](#_Toc122378056)

[Hình 3. 14 Sơ đồ chân của LCD 42](#_Toc122378057)

[Hình 3. 15 Băng tải mini 44](#_Toc122378058)

[Hình 3. 16 Cấu tạo băng tải 45](#_Toc122378059)

[Hình 3. 17 Sơ đồ nguyên lý mạch đầu ra 5V 47](#_Toc122378060)

[Hình 3. 18 Sơ đồ đấu nối với cảm biến 48](#_Toc122378061)

[Hình 3. 19 Khối xử lý trung tâm sử dụng Ardunio Uno R3 48](#_Toc122378062)

[Hình 3. 20 Chu kì hoạt động của động cơ servo 49](#_Toc122378063)

[Hình 3. 21 Sơ đồ nối chân LCD với Ardunio 50](#_Toc122378064)

[Hình 3. 22 Sơ đồ nguyên lý toàn mạch 51](#_Toc122378065)

[Hình 3. 23 Phần mềm proteus 52](#_Toc122378066)

[Hình 3. 24 Sơ đồ mạch nguyên lý 53](#_Toc122378067)

[Hình 3. 25 Mạch in PCB 54](#_Toc122378068)

[Hình 3. 26 Quy trình làm việc của ardunio 55](#_Toc122378069)

[Hình 3. 27 Giao diện lập trình của Ardunio 55](#_Toc122378070)

[Hình 3. 28 Giao diện menu Ardunio IDE 56](#_Toc122378071)

[Hình 3. 29 Giao diện file menu Ardunio IDE 56](#_Toc122378072)

[Hình 3. 30 Giao diện edit menu Ardunio IDE 56](#_Toc122378073)

[Hình 3. 31 Giao diện sketch menu Ardunio IDE 57](#_Toc122378074)

[Hình 3. 32 Giao diện tools menu Ardunio IDE 57](#_Toc122378075)

[Hình 3. 33 Mạch sau khi in thành công 59](#_Toc122378076)

[Hình 3. 34 Lưu đồ giải thuật hệ thống 60](#_Toc122378077)

[Hình 4. 1 Sản phẩm hoàn thiện 65](#_Toc122378078)

# CHƯƠNG 1: TỔNG QUAN

## 1. Lý do chọn đề tài

Khoa học kỹ thuật luôn phát triển trong tất cả các lĩnh vực, nhất là các ngành sản xuất. Việc đòi hỏi cải tiến và nâng cấp hệ thống sản xuất luôn là ưu tiên hàng đầu. Một trong những cải tiến và nâng cấp đó là tự động hóa hệ thống phân loại sản phẩm. Hệ thống này giúp cho sản xuất được linh hoạt hơn, tiết kiệm thời gian và nhân lực, tăng năng suất, đem lại lợi ích kinh tế cao và hiệu quả. Công nghiệp đòi hỏi phải ngày càng chính xác cao và nhanh chóng để đáp ứng được xu thế hiện đại hóa. Các ngành công nghiệp đóng gói sản phẩm, dược phẩm cũng như trong lĩnh vực điện, điện tử là những ngành cần sự chính xác trong kiểm tra đầu ra, và để thay thế con người trong việc kiểm tra thành phẩm với một tốc độ và sự chính xác cao, công nghệ băng chuyền di chuyển sản phẩm ra đời và không ngừng phát triển để ngày càng hoàn thiện hơn.

Sự ra đời của băng tải và những ứng dụng của nó rất cần thiết để phục vụ nhu cầu trong việc sản xuất công nghiệp và từng bước hiện đại hóa ngành công nghiệp. Lịch sử ra đời của băng tải bắt đầu trong nửa sau của thế kỷ 17. Kể từ đó, băng tải là một phần tất yếu của vận chuyển vật liệu. Vào năm 1795, băng tải đã trở thành một phương tiện phổ biến để chuyển tải vật liệu rời, ban đầu nó chỉ được sử dụng để vận chuyển bao tải hạt với khoảng cách ngắn.

Hệ thống băng chuyền có một chiếc giường bằng gỗ phẳng và một vành đai đi qua chiếc giường bằng gỗ. Trước đó, băng tải được làm bằng da, vải hay cao su, hệ thống băng chuyền này đã rất phổ biến cho việc chuyên chở hàng cồng kềnh từ nơi này sang nơi khác. Vào đầu thế kỷ 20, các ứng dụng của băng tải được sử dụng rộng rãi hơn.

Hymle Goddard là người đầu tiên được nhận bằng sáng chế cho các băng tải con lăn trong năm 1908. Các doanh nghiệp con lăn băng tải không phát triển thịnh vượng. Một vài năm sau đó, vào năm 1919 cung cấp và băng tải tự do được đưa vào trong việc sản xuất ô tô. Như vậy, băng tải đã trở thành công cụ phổ biến cho việc chuyên chở hàng hóa nặng và lớn trong các nhà máy.



Hình 1. Hệ thống băng tải vận chuyển đá những năm 90

Trong thập niên 1920 đã được phổ biến và cũng đã trải qua những thay đổi to lớn. Băng tải được sử dụng trong các mỏ than để xử lý chạy than hơn 8km, đã được sử dụng bằng cách dùng lớp bông và cao su bìa. Các băng tải dài nhất hiện nay được sử dụng là dài hàng trăm km ở các khu mỏ phosphate Tây Sahara.

Một trong những bước ngoặt trong lịch sử của băng tải là sự ra đời của băng tải tổng hợp. Nó được giới thiệu trong chiến tranh thế giới thứ 2, chủ yếu là do sự khan hiếm của các vật liệu tự nhiên như bông, cao su.. Kể từ đó, băng tải tổng hợp đã trở nên phổ biến ở các lĩnh vực khác nhau.

Ở nước ta hiện nay, lĩnh vực sử dụng băng tải để vận chuyển sản phẩm phát triển rất đáng kể. Tuy nhiên nhìn một cách tổng quan thì số lượng các ứng dụng trên thực tế vẫn là quá ít, lĩnh vực này sẽ còn phát triển mạnh mẽ trong tương lai nếu được quan tâm một cách nghiêm túc.

Xuất phát từ mục tiêu tiếp cận, bổ sung các kiến thức mới, cũng như củng cố

lại những kỹ năng kiến thức trong suốt quá trình học tập tại trường. Đồng thời,

nghiên cứu sâu hơn về ứng dụng của vi điều khiển, nghiên cứu cách hoạt động của băng tải. Do đó, em thực hiện đồ án điện tử chọn đề tài “ Phân loại sản phẩm theo chiều cao ”.

## 2. Mục tiêu đề tài

Đề tài là bước đầu tìm hiểu những ứng dụng của Ardunio trong thực tế, đồng thời cũng là bước triển khai những kiến thức đã được học vào việc thiết kế và thi công hệ thống phân loại sản phẩm. Thông qua việc nghiên cứu và làm việc nghiêm túc để rèn luyện tác phong, cũng như hoàn thiện phương pháp, tư duy nghiên cứu, giải quyết một vấn đề. Với những mục tiêu đề tài hướng đến là:

• Tìm hiểu những kiến thức về vi điều khiển Ardunio

• Tìm hiểu kiến thức và cách hoạt động của băng tải

• Tìm hiểu về động cơ servo và các loại cảm biến

• Xây dựng chương trình phân loại sản phẩm

## 3. Phương pháp nghiên cứu

Nghiên cứu lý thuyết để tổng hợp và phân tích tài liệu có liên quan từ đó xây dựng cơ sở lí luận cho đề tài.

Sử dụng các công thức toán học để tính toán các phần tử của mạch, xử lí số liệu đo được thông qua thực nghiệm nhằm rút ra các kết luận chính xác.

## 4. Giới hạn đề tài

* Chỉ dùng 2 cảm biến quang nên chỉ phân biệt được 2 loại sản phẩm là thấp và cao.
* Chỉ là mô hình chưa thể mang vào thực tế.
* Cảm biến có thể bị ảnh hưởng do điều kiện môi trường.
* Chưa có giám sát mô hình trên máy tính.
* Độ ổn định chưa được tối ưu.

## 5.Bố cục đề tài

Với các yêu cầu về nhiệm vụ và mục tiêu đề ra, báo cáo đồ án điện tử bao gồm nội dung của các chương như sau:

* CHƯƠNG 1: Tổng Quan

Đặt vấn đề tổng quan về đề tài, tìm hiểu những lý do và sự cần thiết để thực hiện đề tài, mục tiêu hoàn thành, giới hạn cũng như những bước đi từ cơ bản đến cụ thể mà em sẽ thực hiện trong quá trình nghiên cứu đề tài.

* CHƯƠNG 2: Cơ Sở Lý Thuyết

Trình bày các kiến thức cơ bản về Ardunio, các loại cảm biến và cách thức hoạt động của động cơ servo trong đề tài.

* CHƯƠNG 3: Thiết Kế Hệ Thống

Trình bày sơ đồ khối của hệ thống, thiết kế cho từng khối. Sơ đồ kết nối phần cứng, nguyên lý hoạt động.

* CHƯƠNG 4: Thi Công Hệ Thống

Trình bày về các chương trình, cách thức hoạt động của hệ thống và xây dựng bài toán phân loại sản phẩm trên Ardunio Uno R3.

* CHƯƠNG 5: Kết Quả, Nhận Xét và Đánh Giá

Trình bày kết quả đạt được, đưa ra nhận xét và đánh giá những gì đã đạt được và chưa đạt so với mục tiêu đề ra ban đầu.

* CHƯƠNG 6: Kết Luận và Hướng Phát Triển

Kết Luận chung về đề tài, hướng phát triển của đề tài để cải thiện và đáp ứng nhu cầu cuộc sống hiện nay.

# CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

## 2.1. Giới thiệu về hệ thống đếm và phân loại sản phẩm

Ngày nay, cùng với sự tiến bộ của khoa học kĩ thuật, công nghệ điện tử đã, đang và sẽ phát triển ngày càng rộng rãi, trong đó có thể nói đến là vi điều khiển. Các ứng dụng của vi điều khiển như làm Robot, máy bay không người lái, điều khiển tín hiệu đèn giao thông, làm hiệu ứng đèn led nhấp nháy trên các bảng quảng cáo, điều khiển các thiết bị cảm ứng ánh sáng, âm thanh,… Trong các công ty xí nghiệp, ứng dụng hệ thống đếm và phân loại sản phẩm rất phổ biển.

Băng tải được ứng dụng và lắp đặt trong các quy trình sản xuất của nhiều ngành nghề như khai thác khoáng sản, chế biến thực phẩm, đồ uống dược phẩm, phân loại rác thải, ứng dụng trong các thiết bị nâng và bốc dỡ vật liệu kết hợp phân loại sản phẩm,… Trong đó phần lớn các tủ điều khiển hệ thống băng tải để đếm và phân loại sản phẩm còn khá lớn và phức tạp, để giải quyết vấn đề này ta có thể áp dụng vi điều khiển.

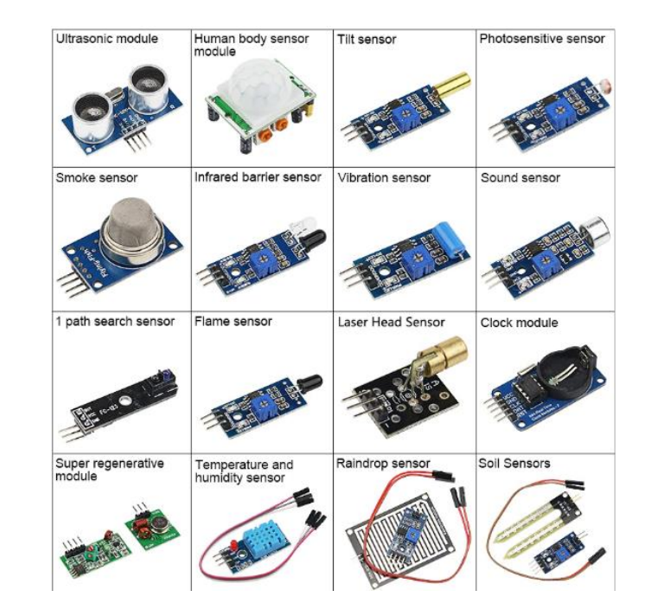
Mô hình phân loại sản phẩm này hoạt động trên nguyên lý dùng các cảm biến để xác định chiều cao của sản phẩm. Sau đó dùng Servo để gạt các sản phẩm có cùng kích thước vào cùng một nơi chứa, đồng thời ghi nhận lại số lượng của mỗi loại sản phẩm vừa phân loại được.

## 2.2 Tổng quan về cảm biến

### 2.2.1 Giới thiệu chung về cảm biến

Cảm biến là một trong những thiết bị điện tử có chức năng cảm nhận những trạng thái hay quá trình vật lý, hóa học sinh học trong môi trường cần khảo sát để biến đổi thành tín hiệu nhằm thu thập dữ liệu về trạng thái, quá trình đó.

Thông tin thu thập sẽ được xử lý để rút ra tham số định tính hoặc định lượng của môi trường nhằm phục vụ các nhu cầu nghiên cứu khoa học hay còn được gọi là đo đạc, phục vụ trong truyền và xử lý thông tin hay trong điều khiển của các quá trình khác. Bộ cảm biến thường được đặt trong các vỏ bảo vệ để tạo thành đầu thu hay đầu dò, kèm theo các mạch điện hỗ trợ, và trong nhiều trường hợp trọn bộ đó lại được gọi luôn là “cảm biến”



Hình 2. Một số loại mạch cảm biến

* + 1. Phân loại các cảm biến

Trên thực tế, hiện nay có rất nhiều loại cảm biến khác nhau, chúng ta có thể chia các loại cảm biến thành hai nhóm chính:

* Cảm biến vật lý: Có thể kể đến một vài ví dụ dễ hình dung như sóng điện từ, ánh sáng, hồng ngoại, tia X, hạt bức xạ, nhiệt độ, áp suất, âm thanh, từ trường, gia tốc,….



Hình 2. Cảm biến hồng ngoại

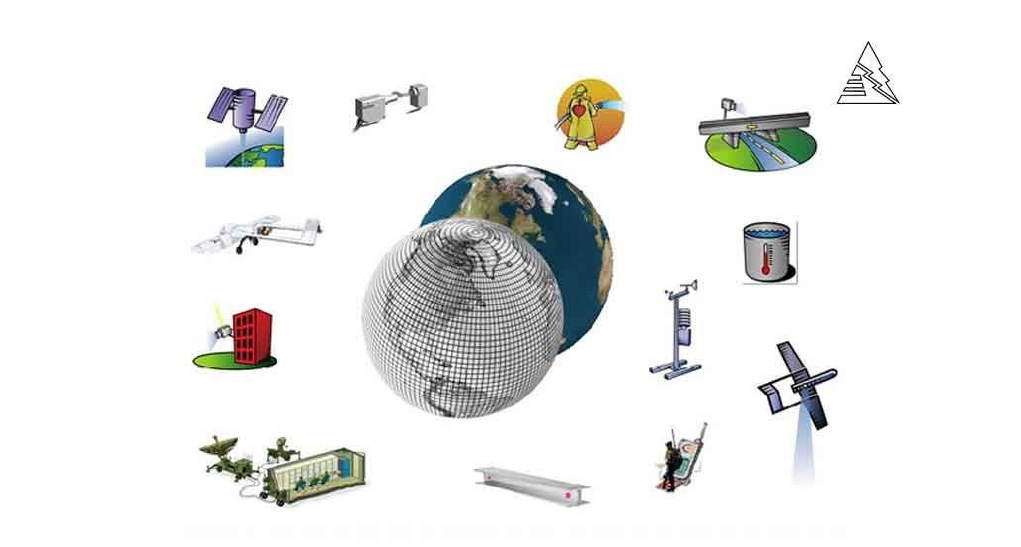
* Cảm biến hóa học: Cảm biến hóa học thường thấy như độ ẩm, độ PH, ion, khói,…

Ngoài ra ta cũng có thể chia các loại cảm biến theo nguyên lý hoạt động, có thể kể đến những loại cảm biến nổi bật sau:

* Cảm biến điện trở: Hoạt động dựa theo di chuyển của con chạy hoặc góc quay của biến trở, hoặc sự thay đổi điện trở do co giãn vật chất.
* Cảm biến cảm ứng: Cảm biến biến áp vi phân, cảm biến cảm ứng điện tử, cảm biến dòng xoáy, cảm biến cảm ứng điện động, cảm biến điện dung,..
* Cảm biến điện trường: Cảm biến từ giảo, cảm biến áp điện,…
* Và một số cảm biến nổi bật khác như: Cảm biến quang, cảm biến điện hóa đầu dò ion, độ PH, cảm biến nhiệt độ,…
  + 1. Vai trò của cảm biến

Đối với các bài toàn tự động hóa nói chung và điều khiển nói riêng thì tự động hóa vô cùng quan trọng.

* Cảm biến giúp nhận diện các tín hiệu đầu vào và đầu ra của hệ thống.
* Cảm biến giúp đo đạc, tính toán các giá trị hệ thống, sản phẩm.
* Cảm biến giới hạn cảm nhận với đại lượng vật lý cần đo.

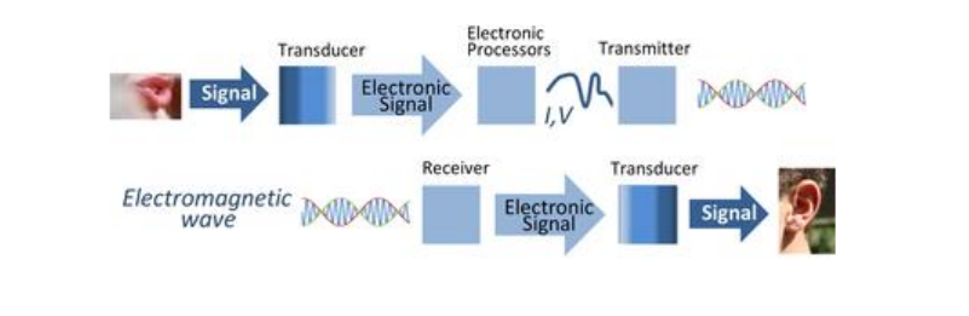


Hình 2. Ứng dụng của cảm biến

* 1. Tổng quan về xử lý tín hiệu

2.3.1 Khái niệm xử lý tín hiệu

* Tín hiệu là biểu hiện vật lý của thông tin. Về mặt toán học tín hiệu được coi là hàm của một hàm hay nhiều biến độc lập, tín hiệu mang thông tin của diễn biến hay bản chất của một hiện tượng.
* Xử lý tín hiệu là quá trình phân tích, sửa đổi và tổng hợp các tín hiệu nhằm cải thiện đường dẫn, hiệu quả và chất lượng của thiết bị cần xử lý.

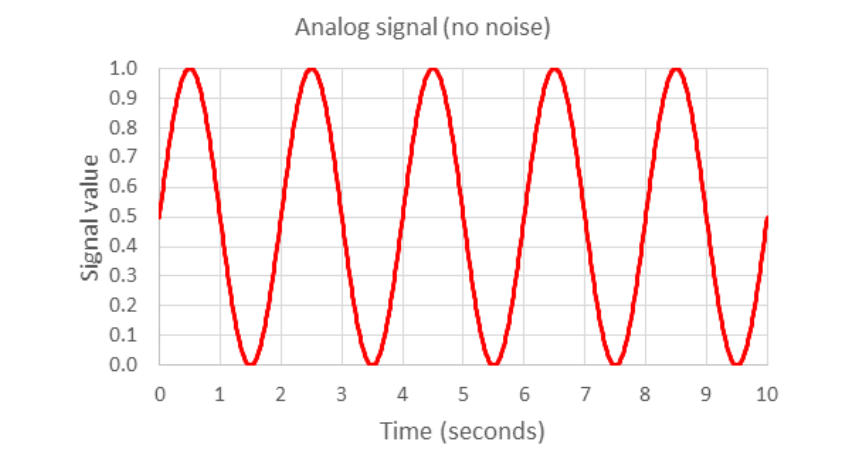


Hình 2. Xử lý tín hiệu

### 2.3.2 Phân loại xử lý tín hiệu

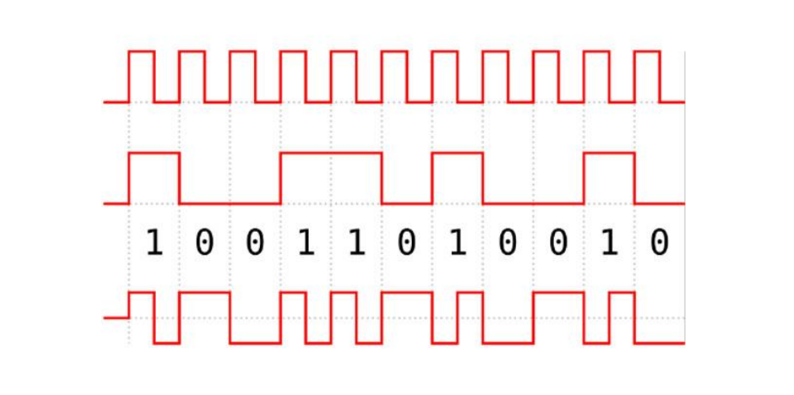
Xử lý tín hiệu chia làm hai loại: Xử lý tín hiệu tương tự và xử lý tín hiệu số.

* Xử lý tín hiệu tương tự: Xử lý tín hiệu tương tự dành cho các tín hiệu chưa được số hóa như trong hầu hết các hệ thống radio, điện thoại, radar, truyền hình thế kỷ 20. Điều này liên quan đến các mạch điện tử tuyến tính cũng như các mạch điện tử phi tuyến.



Hình 2. Tín hiệu tương tự

* Xử lý tín hiệu số: Xử lý tín hiệu số là xử lý các tín hiệu được lấy mẫu thời gian rời rạc số hóa. Việc xử lý được thực hiện bởi các máy tính đa năng hoặc bởi các mạch kĩ thuật số như ASIC, bộ xử lý tín hiệu chuyên dụng,…



Hình 2. Tín hiệu số

* Xử lý tín hiệu thời gian liên tục: Xử lý tín hiệu thời gian liên tục là dành cho các tín hiệu thay đổi theo sự thay đổi của miền liên tục. Các phương pháp xử lý tín hiệu bao gồm miền thời gian, miền tần số và miền tần phức.
* Xử lý tín hiệu thời gian rời rạc: Xử lý tín hiệu thời gian rời rạc là dành cho các tín hiệu được lấy mẫu, chỉ được xác định tại các điểm riêng biệt theo thời gian và như vậy được lượng tử hóa theo thời gian, nhưng không phải là cường độ.
* Xử lý tín hiệu phi tuyến: Xử lý tín hiệu phi tuyến bao gồm việc phân tích và xử lý các tín hiệu được tạo ra từ các hệ thống phi tuyến và có thể thuộc các miền thời gian, tần số hoặc không gian thời gian.
* Xử lý tín hiệu thống kê: Xử lý tín hiệu thống kế là một cách tiếp cận coi tín hiệu là các quá trình ngẫu nhiên, sử dụng các thuộc tính thống kế của chúng để thực hiện các nhiệm vụ xử lý tín hiệu.
  1. Khái quát về vi điều khiển
     1. Giới thiệu về vi điều khiển

Vi điều khiển là một máy tính được tích hợp trên một chíp, nó thường được sử dụng để điều khiển các thiết bị điện tử.

Vi điều khiển, thực chất, là một hệ thống bao gồm một vi xử lý có hiệu suất đủ dùng và giá thành thấp (khác với các bộ vi xử lý đa năng dùng trong máy tính) kết hợp với các khối ngoại vi như bộ nhớ, các mô đun vào/ra, các mô đun biến đổi số sang tương tự và tương tự sang số,…

Ở máy tính thì các mô đun được xây dựng bởi các chip và mạch ngoài.



Hình 2. Vi điều khiển

Một vi điều khiển có kết cấu gồm 2 đầu: đầu vào và đầu ra; bao gồm các cuộn dây, màn hình LCD, rơ le, chuyển mạch và cảm biến… nhằm cung cấp các dữ liệu như độ ẩm, nhiệt độ hay mức độ ánh sáng, mối tương tác, liên kết giữa những vi điều khiển với nhau.

Các vi điều khiển có thể được cấu hình với trạng thái để một đầu vào hoặc một phần mềm đầu ra. Ở đầu vào, các chân đầu vào có thể được sử dụng để đọc các tín hiệu từ bên ngoài hoặc các cảm biến tác động lên vi xử lý. Trong khi đó, các chân đầu ra được kết nối với thiết bị hiển thị kết quả bên ngoài như màn hình LED và động cơ.

* + 1. Các loại vi điều khiển và ứng dụng thực tiễn

Có rất nhiều loại vi điều khiển được lập trình khác nhau, chủ yếu chúng được phân loại và lập trình chuyên sâu theo một số thông số cơ bản, bao gồm Bits, kích thước Flash, kích thước bộ nhớ RAM, số lượng các dòng đầu vào / đầu ra, loại bao bì, cung cấp điện áp và tốc độ. Người dùng có khả năng tinh chỉnh các thông số kỹ thuật cần thiết trong bộ lọc tham số để vi điều khiển có thể cung cấp đúng loại dữ liệu mình cần. Các vi điều khiển đều có thiết kế chung gồm chân đầu vào / đầu ra. Số lượng các chân khác nhau tùy thuộc vào vi điều khiển).

Vi điều khiển vốn được lập trình để sử dụng cho các ứng dụng nhúng, không giống như các bộ vi xử lý trong máy tính cá nhân.

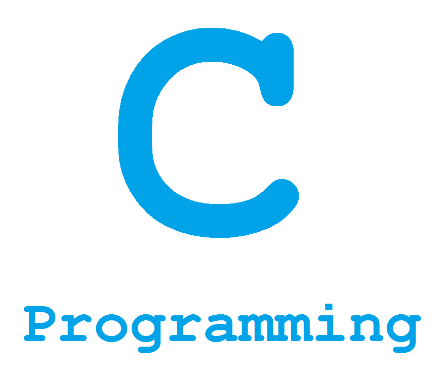
Vi điều khiển thường được sử dụng trong các thiết bị điều khiển tự động bao gồm các công cụ điện, đồ chơi, thiết bị y tế cấy dưới da, máy móc văn phòng, hệ thống điều khiển động cơ, thiết bị, điều khiển từ xa và hàng loạt các hệ thống nhúng khác.



Hình 2. Vi điều khiển dùng trong Ardunio

* 1. Tìm hiểu ngôn ngữ C
     1. Lịch sử hình thành ngôn ngữ C

C là một ngôn ngữ lập trình cấp trung, ngôn ngữ lập trình C được phát triển bởi Dennis M. Ritchie để phát triển hệ điều hành UNIX tại Bell Labs. C được thực thi lần đầu tiên trên máy tính DEC PDP-11 vào năm 1972.



Hình 2. Ngôn ngữ lập trình C

C là một ngôn ngữ lập trình phổ biến, dễ sử dụng, đơn giản và linh hoạt. Nó là một ngôn ngữ lập trình có cấu trúc độc lập với máy và được sử dụng rộng rãi để viết các ứng dụng khác nhau: Hệ điều hành như Windows và nhiều chương trình phức tạp khác như cơ sở dữ liệu Oracle, Git, trình thông dịch Python,… Có thể nói rằng, C là ngôn ngữ lập trình bậc nhất, C là cơ sở cho việc lập trình, là tiền đề để nắm bắt kiến thức của các ngôn ngữ lập trình khác dễ dàng hơn.

* + 1. Đặc điểm nổi bật và ứng dụng của ngôn ngữ C
* *Đặc điểm nổi bật của ngôn ngữ C:*

C là ngôn ngữ được sử dụng rộng rãi, cung cấp một vài tính năng dưới đây:

* C là một ngôn ngữ đơn giản theo hướng cung cấp một cách tiếp cận có cấu trúc ( chia vấn đề thành nhiều phần nhỏ ), ngoài ra nó còn có tập hợp phong phú các hàm thư viện, kiểu dữ liệu,…
* Các chương trình lập trình C có thể được thức thi trên các máy khác nhau với một số thay đổi cụ thể của máy. Do đó, C là một ngôn ngữ độc lập máy.
* C được dự định sử dụng lập trình cấp thấp, dùng để phát triển các ứng dụng hệ thống như nhân hệ điều hành, trình điều khiển,… tuy nhiên cũng hỗ trợ các tính năng của ngôn ngữ cấp cao nên được gọi là ngôn ngữ lập trình trung cấp.
* C là ngôn ngữ lập trình theo nghĩa chúng ta chia chương trình thành các phần bằng cách sử dụng hàm, vì thế nó dễ hiểu và dễ chỉnh sửa. Ngoài ra, vì sử dụng các hàm nên code C cũng có khả năng tái sử dụng.
* C cung cấp nhiều thư viện có sẵn giúp việc phát triển ứng dụng trở nên nhanh chóng hơn.
* Trong ngôn ngữ C, chúng ta có thể giải phóng một bộ nhớ được cấp phát tại thời điểm bất kỳ bằng cách gọi hàm Free().
* Thời gian biên dịch và thực thi của ngôn ngữ C là nhanh do các hàm sẵn có ít hơn vì thế đỡ tốn tài nguyên hơn.
* C cung cấp tính năng con trỏ, chúng ta có thể tương tấc trực tiếp với bộ nhớ bằng việc sử dụng con trỏ và có thể sử dụng con trỏ cho bộ nhớ, cấu trúc, hàm, mảng,…
* Trong C, chúng ta có thể gọi hàm ngay bên trong bản thân hàm đó. Nó cung cấp những mã có khả năng sử dụng lại cho mỗi hàm.
* Với những tính năng nổi bật trên ngôn ngữ C là ngôn ngữ được sử dụng rộng rãi nhất trong các hệ điều hành và phát triển hệ thống nhúng hiện nay.

*Các kiểu mã thường được sử dụng trong C.*

|  |  |  |
| --- | --- | --- |
| Kiểu dữ liệu | Kích thước (byte) | Phạm vi giá trị |
| Unsigned char | 1 | 0 đến 255 |
| Signed char | 1 | -128 đến 128 |
| Unsigned int | 4 | -2,147,483,648 đến 2,147,483,647 |
| Signed int | 4 | 0 đến 4,294,967,295 |
| Short int | 2 | -32,768 đến 32,767 |
| Unsigned short int | 2 | 0 đến 65,535 |
| Long int | 4 | -2,147,483.648 đến 2,147,483,647 |
| float | 4 | 1.2E - 38 đến 3.4E + 38 |
| double | 8 | 2.3E – 308 đến 1.7E + 308 |
| Long double | 10 | 3.4E – 4932 đến 1.1E + 4932 |

Bảng 2. Các kiểu mã trong C

* *Ứng dụng của lập trình C*

Ngôn ngữ C không chỉ linh hoạt trong việc ứng dụng giúp định hình nên các hệ điều hành Windows, Unix và Linux mà nó còn có thể làm việc hiệu quả trong nhiều công việc. Ví dụ như lập trình game, đồ họa và một số ứng dụng tính toán.

* Ngôn ngữ lập trình C có thể được sử dụng để thiết kế phần mềm hệ thống như là hệ điều hành và trình biên dịch. Mục đích chính của việc tạo ra C là viết kịch bản hệ điều hành Unix. Ngôn ngữ C là một phần không thể thiếu trong quá trình phát triển của nhiều hệ điều hành như Unix-kernel, các tiện ích và ứng dụng hệ điều hành Microsoft Windows và một bộ phận lớn hệ điều hành Android.
* Ngôn ngữ C là cơ sở để phát triển các ngôn ngữ mới, ảnh hường trực tiếp hoặc gián tiếp đến sự phát triển của nhiều ngôn ngữ bao gồm C++, C#, Java,Python, Verilog,…Các ngôn ngữ này dựa trên C và biến đổi.
* Ngôn ngữ C thực hiện các thuật toán và cấu trúc dữ liệu nhanh chóng. Tạo điều kiện cho việc tính toán nhanh hơn trong các chương trinh, điều này cho phép sử dụng C trong các ứng dụng yêu cầu mức độ tính toán cao hơn như Matlab và Mathematica.
* Với các tính năng của ngôn ngữ C như là truy cập trực tiếp vào API phần cứng của máy, sự hiện diện của trình biên dịch C, sử dụng tài nguyên xác định và phân bổ bộ nhớ động, đã làm cho ngôn ngữ C trở thành lựa chọn tối ưu cho các ứng dụng và trình điều khiển của các hệ thống nhúng.
* Ngoài các ứng dụng đã được nêu trên, ngôn ngữ C còn được dùng trong đồ họa và lập trình game, sử dụng để phát triển các ứng dụng đồ họa và chơi game, như cờ vua, bắn cung,…

# CHƯƠNG 3: THIẾT KẾ VÀ THI CÔNG HỆ THỐNG

## 3.1 Giới thiệu hệ thống

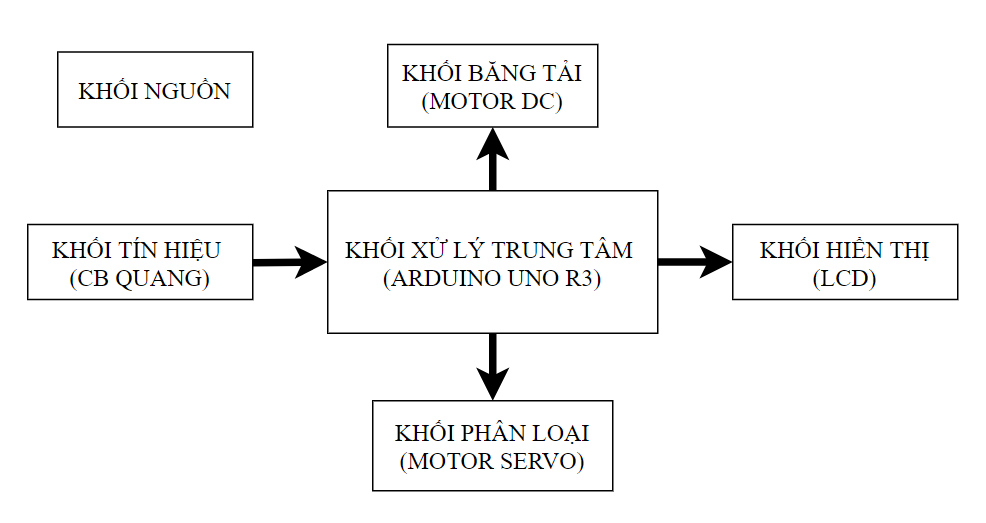
Khoa học kỹ thuật luôn luôn phát triển trong tất cả các lĩnh vực, nhất là các ngành sản xuất. Việc đòi hỏi cải tiến và nâng cấp hệ thống sản xuất luôn là ưu tiên hàng đầu. Một trong những hệ thống đó là hệ thống phân loại sản phẩm tự động. Hệ thống này giúp cho sản xuất linh hoạt hơn, tiết kiệm thời gian và nhân lực, tăng sản lượng, đem lại lợi ích kinh tế cao và hiệu quả.

Để phân loại sản phẩm có rất nhiều phương pháp, tuy nhiên hiện nay phương pháp phân loại theo chiều cao chưa được ứng dụng nhiều và hiệu quả. Do đó, đề tài “ Thiết kế và thi công hệ thống phân loại sản phẩm theo chiều cao ” là một đề tài mang tính nghiên cứu và ứng dụng cao, phù hợp với sự phát triển của các ngành sản xuất.

* 1. Thiết kế sơ đồ khối

` Để giảm thiểu sai sót và không xảy ra sự cố khi thi công mô hình theo ý tưởng đã để ra, ta bắt buộc phải tính toán và thiết kế. Với đề tài: “ Thiết kế và thi công hệ thống phân loại sản phẩm theo chiều cao ” mục tiêu chính là phân loại được từng sản phẩm theo kích thước khác nhau vào từng hộp thì công việc tính toán, thiết kế bao gồm các bước sau:

* Thiết kế sơ đồ khối của hệ thống
* Tính toán, thiết kế từng khối của hệ thống:
* Khối nguồn.
* Khối băng tải.
* Khối tín hiệu.
* Khối xử lý trung tâm
* Khối phân loại
* Khối hiển thị



Hình 3. Sơ đồ khối mô hình phân loại sản phẩm

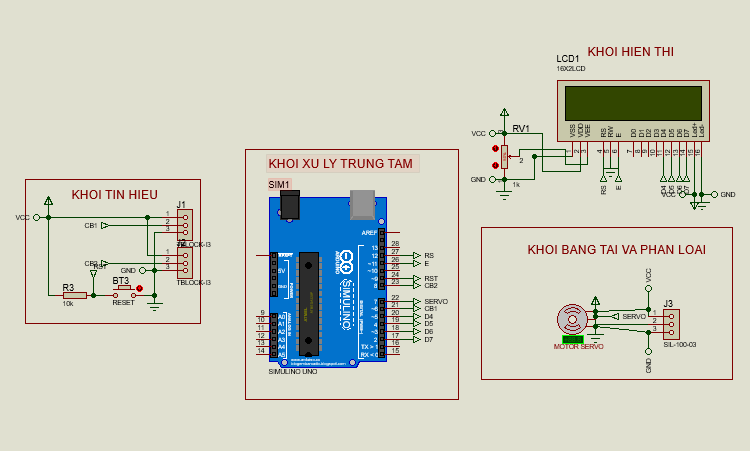
### 3.2.2. Chức năng của từng khối

* Khối nguồn: Cấp nguồn cho toàn bộ hệ thống hoạt động.
* Khối băng tải: Vận chuyển vật thể đến vị trí của cảm biến để nhận dạng.
* Khối tín hiệu: Phát hiện vật thể và truyền tín hiệu về khối xử lý để mã hóa dữ liệu.
* Khối xử lý trung tâm: Xử lý tín hiệu từ cảm biến và xuất dữ liệu được mã hóa đến các khối hiển thị và phân loại.
* Khối hiển thị: Hiển thị số lượng đếm được từ cảm biến
* Khối phân loại: Phân loại các sản phẩm thành nhiều loại theo yêu cầu của đề tài.
  1. Thiết kế sơ đồ nguyên lý

Danh sách các linh kiện:

|  |  |  |
| --- | --- | --- |
| STT | Tên linh kiện | Số lượng |
| 1 | Ardunio Uno R3 | 1 |
| 2 | Adapter | 1 |
| 3 | LCD | 1 |
| 4 | Băng tải | 1 |
| 5 | Cảm biến hồng ngoại | 2 |
| 6 | Động cơ servo | 2 |

Bảng 3. Danh sách các linh kiện



Hình 3. Sơ đồ nguyên lý của hệ thống

* Nguyên lý hoạt động của hệ thống:

Khi được cấp nguồn, mạch sẽ hoạt động theo chương trình được thiết lập trên Arduino. Chu trình hoạt động của mô hình hệ thống này được bắt đầu từ sự thu nhận tín hiệu của các cảm biến. Các tín hiệu được trung tâm xử lý và xuất các lệnh tương thích cho Servo cùng với LCD để thực hiện theo đúng yêu cầu đề tài.

Ở trạng thái bình trường, cảm biến E18-D80NK cấp xung mức cao cho bộ xử lý trung tâm. Với xung mức cao, thông qua code đã được lập trình, Arduino Uno R3 duy trì hiện trạng. Động cơ Servo giữ nguyên thanh gạc phân loại ở góc 0 độ.

Khi được tác động bởi đối tượng, dòng điện qua trở treo thông qua cảm biến E18-D80NK tới GND. Lúc này, xung mức thấp được cấp cho Arduino Uno R3. Dữ liệu được xử lý và đưa vào LCD làm thay đổi sự hiển thị của LCD. Đồng thời, động cơ Servo quay thanh gạc phân loại một góc 50 độ, đối tượng được đưa vào nơi chứa tương ứng.

Khi nhấn nút Reset, xung mức thấp được cấp vào chân số 9 của Arduino Uno R3. Tất cả dữ liệu đếm được sẽ được reset về giá trị 0 và hiển thị lên LCD.

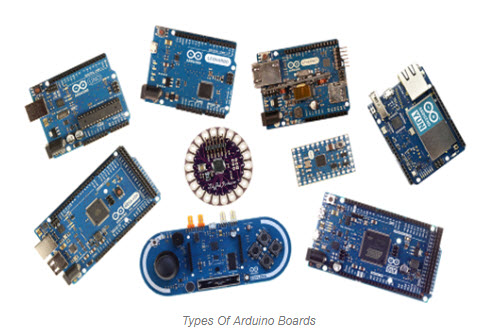
* + 1. Khối xử lý trung tâm

Ardunio được khởi động vào năm 2005 như là một dự án dành cho sinh viên trại Interaction Design Institute tại Ivrea, Italya. Cái tên “ Ardunio” đến từ một quán bar tại Ivera, nơi một vài nhà sáng lập của dự án này thường xuyên gặp mặt.

Các thiết bị dựa trên nền tảng Ardunio được lập trình bằng ngôn ngữ riêng. Ngôn ngữ này dựa trên ngôn ngữ Wiring được viết cho phần cứng nói chung trên một môi trường phát triển tích hợp IDE chạy trên các máy tính cá nhân. Và Wiring lại là một biến thể của C/C++. Một số người gọi nó là Wiring, một số khác gọi là C/C++.

Thông tin thiết kế phần cứng được cung cấp công khai để những ai muốn tự làm một mạch Ardunio bằng tay có thể tự mình thực hiện. Người ta ước tính khoảng giữa năm 2011 có trên 300 ngàn mạch Ardunio chính thức đã được sản xuất thương mại và vào năm 2013 có khoảng 700 ngàn mạch chính thức đã được đưa tới tay người dùng.

Phần cứng Ardunio gốc được sản xuất bởi công ty Italy tên là Smart Projects. Một vài board dẫn xuất từ Ardunio cũng được thiết kế bởi công ty Mỹ tên là SparkFun Electronics. Nhiều phiên bản của Ardunio cũng đã được sản xuất phù hợp cho nhiều mục đích sử dụng.



Hình 3. Các phiên bản Ardunio

Ardunio Uno R3 là một loại bo mạch vi điều khiển dựa trên Atmega328P. Nó bao gồm toàn bộ những thứ cần thiết để giữ bộ vi điều khiển chỉ cần gắn nó vào PC với sự trợ giúp của cáp USB và cung cấp nguồn điện bằng bộ chuyển AC-DC hoặc pin để bắt đầu.

Thuật ngữ Uno có nghĩa là “ một ” trong ngôn ngữ Ý và được chọn để đánh dấu việc phát hành phần mềm Ardunio IDE 1.0.R3, Ardunio Uno là bản sửa đổi thứ 3 cũng như gần đây nhất của Ardunio Uno. Bảng Ardunio và phần mềm IDE là phiên bản tham chiếu của Ardunio và hiện đang được phát hành mới. Bảng mạch Uno là bảng chính trong chuỗi USB.

Board Ardunio Uno R3 được trang bị gồm 1 cổng giao tiếp USB, 6 chân đầu vào Analog, 14 chân I/O tín hiệu số tương thích với nhiều board mở rộng khác nhau, có thể nạp chương trình qua cổng USB, cấp nguồn qua USB hoặc adapter.



Hình 3. Ardunio Uno R3

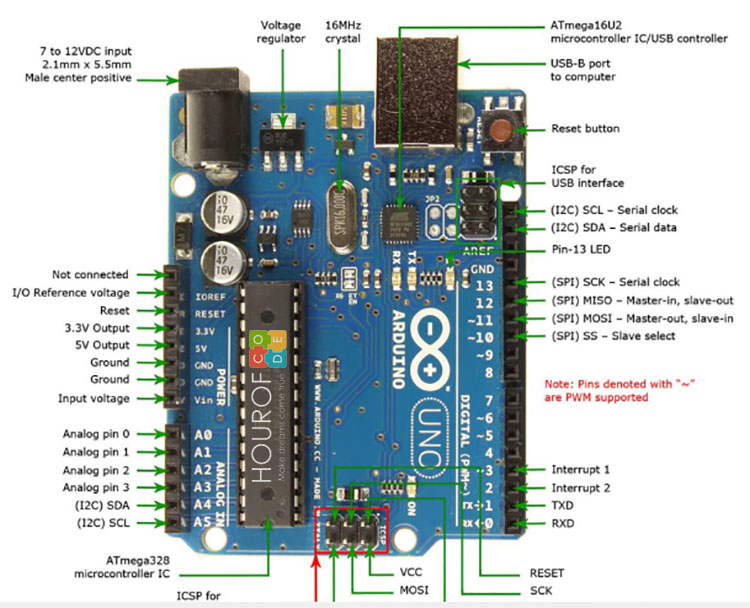
* *Một vài thông số kĩ thuật của Ardunio Uno.*

|  |  |
| --- | --- |
| Vi điều khiển | Atmega328 họ 8bit |
| Điện áp hoạt động | 5V DC ( cấp qua cổng USB ) |
| Điện áp vào khuyên dùng | 7-12V DC |
| Điện áp vào giới hạn | 6-20V DC |
| Tần số hoạt động | 16MHz |
| Dòng điện tiêu thụ | Khoảng 30mA |
| Số chân Digital I/O | 14 ( 6 chân hard ware PWM ) |
| Số chân analog | 6 ( độ phân giải 10bit ) |
| Dòng tối đa trên mỗi chân I/O | 30mA |
| Dòng ra tối đa (5V) | 500mA |
| Dòng ra tối đa (3.3V) | 50mA |
| Bộ nhớ flash | 32kB ( Atmega328 ) với 0.5Kb dùng bởi bootloader |
| SRAM | 2Kb ( Atmega328 ) |
| EEPROM | 1Kb ( Atmega328 ) |
| Xung nhịp | 16MHz |

Bảng 3. Thông số của Ardunio Uno

Thông qua bảng 3.2 chúng ta có cái nhìn tổng quát về Board Ardunio Uno R3 với những thông số chính về điện áp hoạt động, điện áp đầu vào đề nghị và giới hạn, số lượng cũng như dòng tối đa trên các chân I/O, bộ nhớ flash, SRAM, EEPROM, tốc độ của xung đồng hồ. Thông qua những thông số này, chúng ta sẽ có những cách lựa chọn các linh kiện cho phù hợp với những quy định về điện áp, dòng điện, mục đích lập trình để đảm bảo cho mô hình hoạt động ổn định.

* *Vị trí và chức năng các chân của Ardunio*



Hình 3. Sơ đồ chân Ardunio Uno R3

* *Cổng kết nối với Ardunio Uno.*

Ardunio Uno có thể được cấp nguồn 5V thông qua cổng USB hoặc cấp nguồn ngoài với điện áp khuyên dùng 7-12V DC và giới hạn 6-20V DC. Thường thì cấp nguồn bằng pin 9V nếu không có sẵn nguồn từ cổng USB. Nếu cấp nguồn vượt quá ngưỡng giới hạn trên Arduino Uno sẽ hỏng.

Ardunio Uno R3 có 2 chân A4 ( SDA ) và A5 ( SCL ) hỗ trợ giao tiếp I2C/TWI với các thiết bị khác.

* *Lập trình cho Ardunio*

Các thiết bị dựa trên nền tảng Arduino được lập trình bằng ngôn ngữ riêng. Ngôn ngữ này dựa trên ngôn ngữ Wiring được viết cho phần cứng. Và Wiring là một biến thể của C/C++. Một số người gọi đó là Wiring một số khác thì gọi là C hay C++.

Để lập trình cũng như gửi lệnh và nhận tín hiệu từ mạch Arduino, nhóm phát triển dự án này đã cung cấp đến cho người dùng một môi trường lập trình cho Arduino được gọi là Arduino IDE như hình bên dưới.



Hình 3. Giao diện lập trình Ardunio IDE

* + 1. Khối tín hiệu

Cảm biến hồng ngoại ( IR sensor ) là một thiết bị điện tử đo và phát hiện bức xạ hồng ngoại trong môi trường xung quanh. Bức xạ hồng ngoại được phát hiện bởi một nhà thiên văn học tên là William Herchel vào năm 1800. IR sensor là vô hình đối với mắt người, vì bước sóng của nó dài hơn ánh sáng khả kiến, bất kỳ vật nào phát ra nhiệt thì đều phát ra bức xạ hồng ngoại.



Hình 3. Cảm biến hồng ngoại

* *Một vài thông số kĩ thuật và sơ đồ dây của cảm biến hồng ngoại.*

Thông số kĩ thuật cảm biến hồng ngoại:

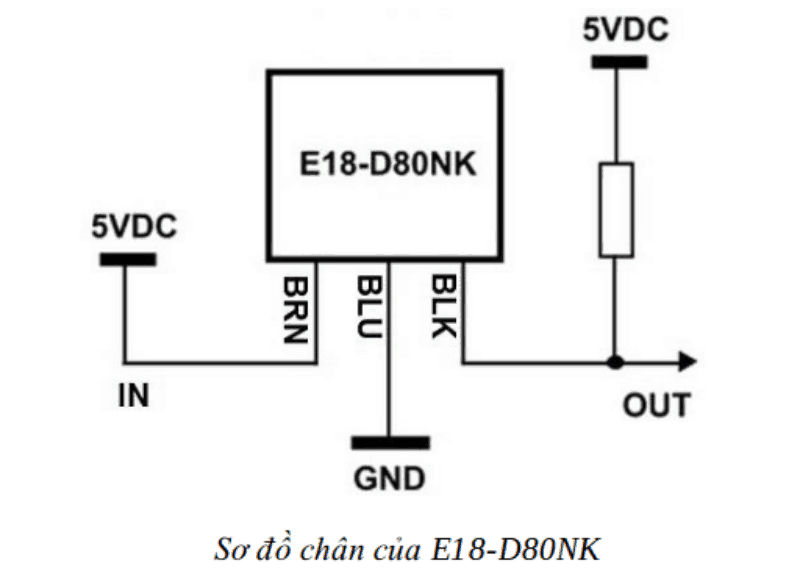
|  |  |
| --- | --- |
| Điện áp hoạt động | 5V DC |
| Khoảng cách hoạt động tối đa | ~80cm |
| Thời gian phản hồi | ~2ms |
| Dòng kích ngõ ra | 300mA |
| Hiện thị ngõ ra | Led đỏ |
| Khả năng phát hiện đối tượng | Trong suốt hoặc đục |
| Nhiệt độ hoạt động | -25 độ C đến 55 độ C |
| Đường kính | 18mm |
| Chiều dài cảm biến | 45mm |
| Chiều dài dây | 45cm |
| Chất liệu vỏ cảm biến | Nhựa |

Bảng 3. Thông số kĩ thuật của cảm biến hồng ngoại

Sơ đồ dây cảm biến hồng ngoại:

Cảm biến hồng ngoại có cách nối dây tương đối đơn giản. Tuy nhiên khi sử dụng cảm biến cần lưu ý những điều sau để trong quá trình sử dụng được thuận tiện hơn:

* Ngõ ra cảm biến dạng NPN cực thu hở giúp tùy biến được điện áp ngõ ra, điện trở kéo lên nguồn bao nhiêu sẽ tạo thành điện áp ngõ ra bấy nhiêu.
* Tín hiệu ra có dòng rất nhỏ nên không được kích trực tiếp vào relay sẽ gây cháy cảm biến.

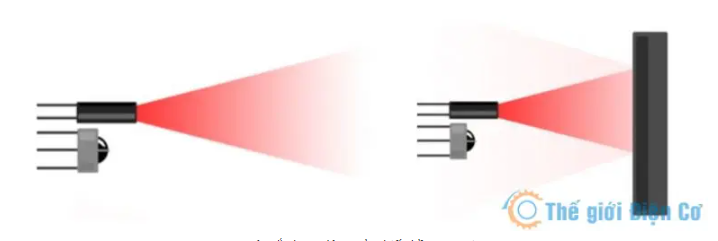


Hình 3. Sơ đồ chân của cảm biến

* Màu nâu: VCC nối với nguồn dương 5V DC.
* Màu xanh dương: GND nối với nguồn âm 0V DC.
* Màu đen: Chân tín hiệu ngõ ra cực thu hở NPM, cần phải có điện trở kéo để tạo thành mức cao.
* *Nguyên lý hoạt động của cảm biến hồng ngoại.*

Cảm biến hồng ngoại sẽ hoạt động bằng cách sử dụng một cảm biến ánh sáng cụ thể để phát hiện bước sóng ánh sáng chọn trong phổ hồng ngoại ( IR ). Bằng cách sử dụng đèn Led tạo ra ánh sáng có cùng bước sóng với cảm biến đang tìm kiếm, chúng ta có thể xem cường độ của ánh sáng nhận được.

Khi một vật ở gần cảm biến, ánh sáng từ đèn LED bật ra khỏi vật thể và đi vào cảm biến ánh sáng. Điều này đẫn dến một bước nhảy lớn về cường độ, mà chúng ta đã biết có thể được phát hiện bằng cách sử dụng một ngưỡng.



Hình 3. Nguyên lý hoạt động của cảm biến

* + 1. Khối phân loại
* Động cơ servo

Động cơ servo được thiết kế cho những hệ thống hồi tiếp vòng kín. Tín hiệu ra của động cơ được nối với một mạch điều khiển. Khi động cơ quay, vận tốc và vị trí sẽ được hồi tiếp về mạch điều khiển. Nếu có bất kỳ lý do nào ngăn cản chuyển động quay của động cơ, cơ cấu hồi tiếp sẽ nhận thấy tín hiệu ra chưa đạt được vị trí mong muốn. Mạch điều khiển tiếp tục chỉnh sai lệch cho động cơ đạt được điểm chính xác.



Hình 3. Động cơ servo

* *Một vài thông số kĩ thuật của động cơ servo*

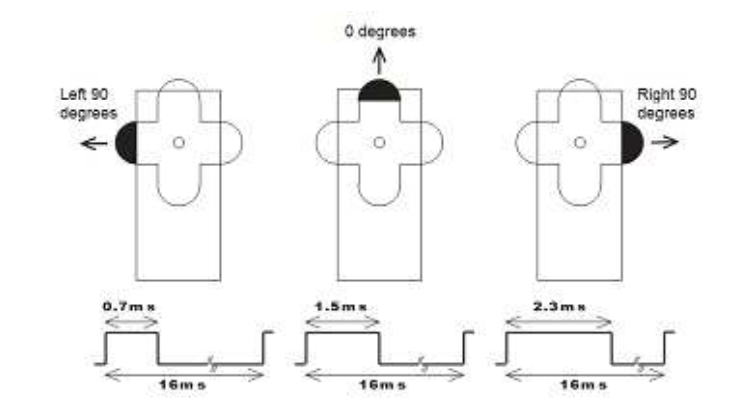
|  |  |
| --- | --- |
| Khối lượng | 55g |
| Kích thước | 40.7 x 19.7 x 42.9mm |
| Momen xoắn tại nguồn 4.8V | 9.4 kgf.cm |
| Momen xoắn tại nguồn 6V | 11 kgf.cm |
| Điện áp hoạt động | 4.8V đến 7.2V |
| Nhiệt độ hoạt động | 0 độ C đến 55 độ C |

Bảng 3. Thông số của động cơ servo

* *Hoạt động của động cơ servo*

Trục của động cơ Servo được định vị nhờ vào kỹ thuật gọi là điều biến độ rộng xung ( PWM ). Trong hệ thống này, servo là đáp ứng của một dãy các xung số ổn định. Cụ thể hơn, mạch điều khiển là đáp ứng của một tín hiệu số có các xung biến đổi từ 1 – 2 ms. Các xung này được gửi đi 50 lần/giây. Chú ý rằng không phải số xung trong một giây điều khiển servo mà là chiều dài của các xung. Servo đòi hỏi khoảng 30 – 60 xung/giây. Nếu số này quá thấp, độ chính xác và công suất để duy trì servo sẽ giảm.

Với độ dài xung 1ms, servo được điều khiển quay theo một chiều ( giả sử là chiều kim đồng hồ như hình dưới đây ).



Hình 3. Hoạt động của động cơ servo

Với độ dài xung 2ms, servo quay theo chiều ngược lại. Kỹ thuật này còn được gọi là tỉ lệ số - chuyển động của servo tỉ lệ với tần số điều khiển.

* *Các giới hạn quay của động cơ servo*

Các servo khác nhau ở góc quay được với cùng tín hiệu 1 – 2ms ( hoặc bất kỳ ) được cung cấp. Nếu ta cố điều khiển servo vượt quá những giới hạn cơ học của nó, hiện tượng này kéo dài hơn vài giây sẽ làm bánh răng của động cơ bị phá hủy.

* Động cơ DC

Động cơ DC (Direct Current) là động cơ điện một chiều có cấu tạo gồm hai dây (dây nguồn và dây tiếp đất). Động cơ phụ thuộc vào lực mà từ trường tạo ra. Khi cấp điện thì động cơ DC sẽ chuyển điện năng thành cơ năng và quay với tốc độ được tính bằng rpm. Tốc độ của động cơ DC được điều khiển bằng cách biến đổi điện áp cung cấp, thay đổi cường độ dòng điện hoặc điều biến độ rộng xung (PWM).



Hình 3. Động cơ DC

* *Một vài thông số kĩ thuật của động cơ DC*

|  |  |
| --- | --- |
| Điện áp định mức | 12V 24V |
| Dòng tối đa | 1.5A |
| Tốc độ quay | 50 vòng/phút với điện áp 12V  150 vòng/phút với điện áp 24V |

Bảng 3. Thông số của động cơ DC

Động cơ trong mô hình dùng để kéo băng tải hoạt động, vận chuyển sản phẩm di chuyển từ đầu băng tải đến cuối băng tải.

* + 1. Khối hiển thị

Có chức năng hiển thị kết quả được nhận từ Khối điều khiển. Sử dụng màn hình LCD 16x2 để hiển thị kết quả.

* *Sơ lược về LCD*

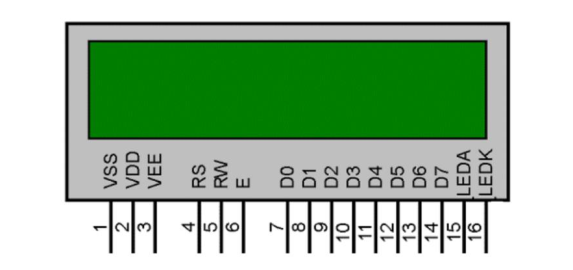
LCD có rất nhiều dạng phân biệt theo kích thước từ vài kí tự đến hàng chục kí tự, từ 1 hàng đến vài chục hàng. Ví dụ LCD 16 x 2 có nghĩa là có 2 hàng, mỗi hàng có 16 ký tự.



Hình 3. LCD 16 x 2

* *Sơ đồ và chức năng các chân của LCD*

LCD có nhiều loại và số chân của chúng cũng khác nhau nhưng có 2 loại phổ biến là loại 14 chân và loại 16 chân. Sự khác nhau là các chân nguồn cung cấp cho đèn nền, còn các chân điều khiển thì không thay đổi.



Hình 3. Sơ đồ chân của LCD

Bảng chức năng chân của LCD:

|  |  |  |
| --- | --- | --- |
| Chân | Ký hiệu | Chức năng |
| 1 | VSS | Chân nối đất cho LCD khi ta thiết kế mạch ta nối chân này với GND của mạch điều khiển. |
| 2 | VDD | Chân cấp nguồn cho LCD, khi thiết kế mạch ta nối chân này với VCC=5V của mạch điều khiển. |
| 3 | VEE | Điều chỉnh độ tương phản của LCD. |
| 4 | RS | Chân chọn thanh ghi (Register select). Nối chân RS với logic “0” (GND) hoặc logic “1” (VCC) để chọn thanh ghi. |
| 5 | RW | Chân chọn chế độ đọc/ghi (Read/Write). Nối chân R/W với logic “0” để LCD hoạt động ở chế độ ghi, hoặc nối với logic “1” để LCD ở chế độ đọc. |
| 6 | E | Chân cho phép (Enable). Sau khi các tín hiệu được đặt lên bus DB0-DB7, các lệnh chỉ được chấp nhận khi có 1 xung cho phép của chân E. |
| 7 - 14 | D0 – D7 | Tám đường của bus dữ liệu dùng để trao đổi thông tin với MPU. |
| 15 | A | Nguồn dương cho đèn nền. |
| 16 | K | GND cho đèn nền |

Bảng 3. Chức năng chân của LCD

Trong 16 chân của LCD, được chia làm 4 dạng tín hiệu dưới đây:

* Các chân cấp nguồn: Chân số 1 là chân nối mass (0V), chân thứ 2 là Vdd nối với nguồn +5V. Chân thứ 3 dùng để chỉnh contrast thường nối với biến trở, chỉnh cho đến khi thấy được kí tự thì ngừng, trong bộ thực hành thì đã chỉnh rồi.
* Các chân điều khiển: Chân số 4 là chân RS dùng để điều khiển lựa chọn thanh ghi. Chân R/W dùng để điều khiển quá trình đọc và ghi. Chân E là chân cho phép dạng xung chốt.
* Các chân dữ liệu D7÷D0: Chân số 7 đến chân số 14 là 8 chân dùng để trao đổi dữ liệu giữa thiết bị điều khiển và LCD.
* Các chân LED\_A và LED\_K: Chân số 15, 16 là 2 chân dùng để cấp nguồn cho đèn nền để có thể nhìn thấy vào ban đêm.
  + 1. Khối băng tải

Băng tải thường được dùng để di chuyển các vật liệu đơn giản và vật liệu rời theo phương ngang và phương nghiêng. Trong các dây chuyền sản xuất, các thiết bị này được sử dụng rộng rãi như những phương tiện để vận chuyển các cơ cấu nhẹ, trong các xưởng luyện kim dùng để vận chuyển quặng, thang đá, các loại xỉ lò trên các trạm thủy điện thì dùng vận chuyển nhiên liệu.

Trên các kho bãi thì dùng để vận chuyển các loại hàng bưu kiện, vật liệu hạt hoặc một số sản phẩm khác. Trong một số nghành công nghiệp nhẹ, công nghiệp thực phẩm, hóa chất thì dùng để vận chuyển các sản phẩm đã hoàn thành và chưa hoàn thành giữa các công đoạn, các phân xưởng, đồng thời cũng dùng để loại bỏ các sản phẩm không dùng được



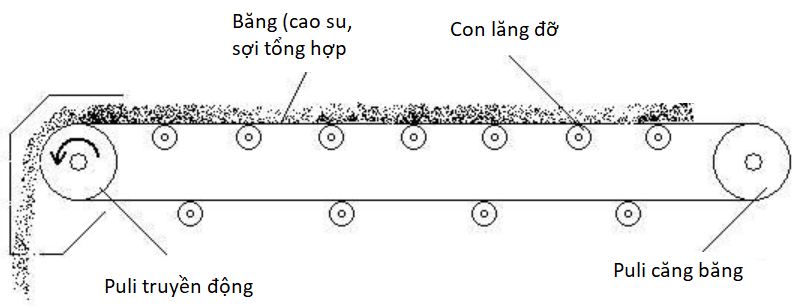
Hình 3. Băng tải mini

#### Ưu điểm của băng tải

Cấu tạo đơn giản, bền, có khả năng vận chuyển rời và đơn chiếc theo các hướng nằm ngang, nằm nghiêng hoặc kết hợp giữa nằm ngang với nằm nghiêng. Vốn đầu tư không lớn lắm, có thể tự động được, vận hành đơn giản, bảo dưỡng dễ dàng, làm việc tin cậy, năng suất cao và tiêu hao năng lượng so với máy vận chuyển khác không lớn lắm.

Tuy vậy phạm vi sử dụng của băng tải bị hạn chế do tốc độ dốc cho phép của băng tải không cao và không đi theo đường cong được.

#### Cấu tạo chung về băng tải



Hình 3. Cấu tạo băng tải

Cấu tạo hệ thống băng tải gồm:

- Khung băng tải: thường được làm bằng nhôm định hình, thép sơn tĩnh điện hoặc inox.

- Băng tải: Thường là dây băng tải cao su lõi bố PE, lõi thép, hoặc băng tải nhựa PVC với các quy cách khác nhau.

- Động cơ: Có các loại động cơ giảm tốc công suất 0.2KW, 0.4KW, 0.75KW, 1.5KW, 2.2KW.

- Bộ điều khiển: Thường gồm biến tần, sensor, timer, PLC…

- Cơ cấu truyền động: Rulo kéo, con lăn đỡ, nhông xích…

### 3.3.6 Tính toán, thiết kế mạch cho từng khối

### Khối nguồn

Trong phạm vi đề tài, chủ yếu sử dụng các linh kiện yêu cầu cấp nguồn 5VDC. Các linh kiện sử dụng nguồn 5V bao gồm: Arduino Uno R3, LCD 16x2, Motor Servo, cảm biến E18-D80NK.

* Chọn IC ổn áp:

Theo datasheet của vi điều khiển ATmega328P, tổng cường độ dòng điện cấp cho các I/O pin tối đa là 300mA. Dòng tiêu thụ tối đa của cảm biến E18-D80NK là 100mA, của servo là 730mA, của LCD là 1mA.

I = 300 + 100x2 + 730 + 1 = 1231 1.23 A

Dựa và các thông số này ta có thể chọn IC ổn áp 7805 loại 1.5A là đủ dòng cấp cho mạch hoạt động.

* Chọn biến áp nguồn:

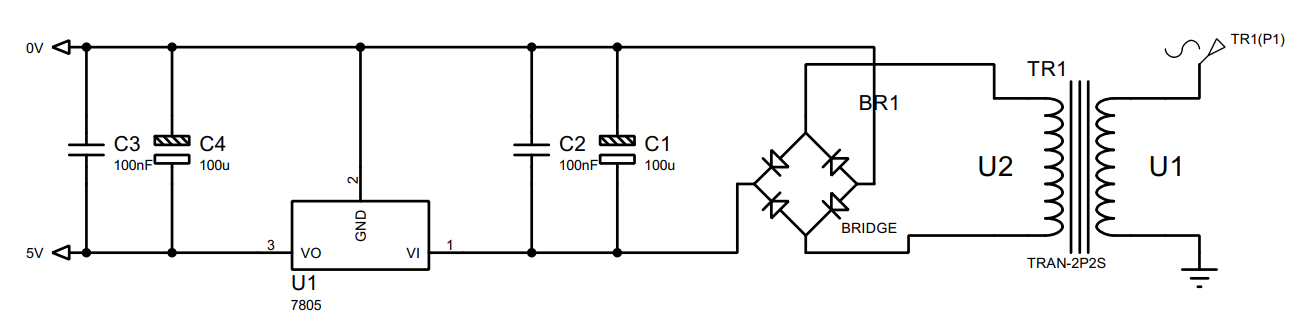
Biến áp là một thiết bị điện gồm 2 cuộn dây quấn trên một lõi sắt từ có nhiệm vụ biến đổi điện áp xoay chiều thành điện áp xoay chiều khác để phù hợp với mục đích cần sử dụng.

Thông thường ra sẽ chọn biến 220V về 12V hoặc 220V về 9V để thiết kế nguồn 5VDC, dòng định mức 1A.

* Chọn mạch chỉnh lưu:

Ta có thể chọn diode rời hoặc cầu diode tích hợp để làm nhiệm vụ biến đổi điện áp xoay chiều thành điện áp một chiều. Cần chọn diode có thông số dòng định mức lớn hơn dòng tải tiêu thụ và điện áp ngược chịu đựng của diode cũng phải lớn hơn điện áp xoay chiều đặt vào diode.

Các diode thông thường hay được sử dụng là: 1N4004, 1N4007, cầu diode 1A, 2A, 5A, 10A, 15A…, và điện áp ngược chịu đựng lên đến 1000V. Dòng tiêu thụ hiện tại của mạch là khoảng 1.23A và điện áp xoay chiều đặt vào diode là 220V nên ta chọn cầu diode 10A và điện áp ngược cực đại 1000V là phù hợp.



Hình 3. Sơ đồ nguyên lý mạch đầu ra 5V

Chỉ duy nhất động cơ DC của băng tải cần cấp nguồn 12VDC nên ta sẽ dùng Adapter để cấp riêng cho động cơ hoạt động.

### Khối tín hiệu

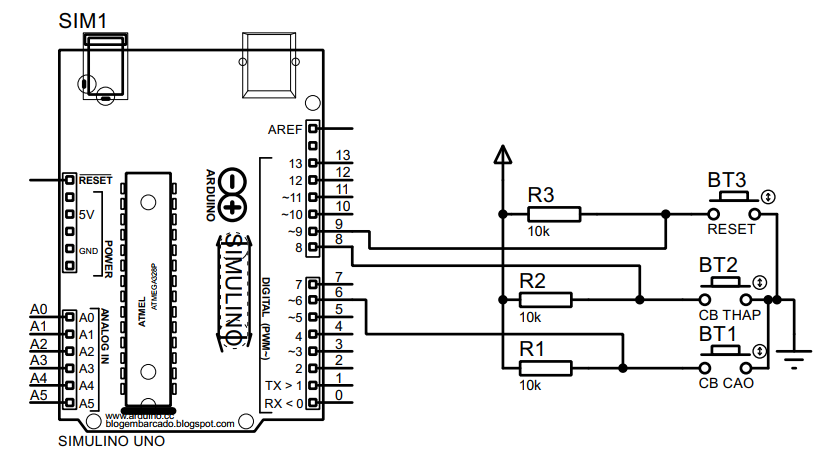
Khối tín hiệu sử dụng 2 cảm biến E18-D80NK nối với các chân 6, 8 và 1 nút nhấn reset nối với chân 9 của Arduino.

Vì cảm biến đã được kéo trở nội 10k lên VCC nên điện áp mức cao của chân tín hiệu (Vout) của cảm biến cũng chính là điện áp VCC, có hai trường hợp như sau:

* Nếu mạch cần giao tiếp sử dụng mức tín hiệu có điện áp bằng với điện áp VCC của cảm biến thì không cần nối trở Rx mà nối trực tiếp chân tín hiệu của cảm biến với chân tín hiệu của Arduino.
* Nếu mạch cần giao tiếp sử dụng mức tín hiệu có điện áp khác với điện áp VCC của cảm biến thì cần nối thêm trở Rx để giá trị của Vout=5VDC không làm cháy mạch Arduino, giá trị của Rx được tính theo công thức cầu phân áp là:

Rx= (Vout\*R1) / (VCC-Vout) (3.1)

Ở đề tài này ta cấp cho cảm biến VCC = 5VDC bằng với điện áp giao tiếp của Arduino là 5VDC nên ta sẽ nối trực tiếp chân tín hiệu của cảm biến với chân tín hiệu của Arduino.



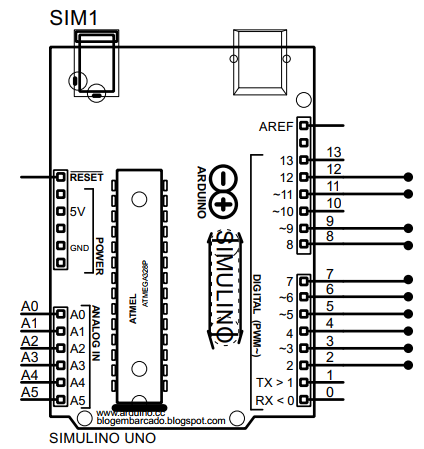
Hình 3. Sơ đồ đấu nối với cảm biến

### Khối xử lý trung tâm

Arduino UNO sử dụng 3 vi điều khiển chính thuộc họ 8bit AVR là ATmega328, ATmega168, ATmega8. Trong quá trình kết nối các module và lập trình cho hệ thống, tổng số chân I/O sử dụng là 10 chân, trong đó có 2 chân nối cảm biến, 1 chân nối nút nhấn reset, 1 chân nối servo và 6 chân nối với LCD, công thức tính dòng tiêu thụ I có thể được tính như sau:

I = P \* I0 =10 \* 20mA = 200mA (3.2)

Trong đó, P là số chân và I0 là dòng ngõ ra mỗi chân I/O.



Hình 3. Khối xử lý trung tâm sử dụng Ardunio Uno R3

### Khối băng tải

Khối băng tải bao gồm 1 động cơ DC 12V, băng tải, khung băng tải và con lăn. Động cơ sẽ được cấp nguồn 12VDC từ Adapter để hoạt động. Ta sử dụng loại băng tải 2 con lăn ở 2 đầu của băng tải, động cơ DC quay kéo theo con lăn ở 1 đầu quay làm cho băng tải chuyển động theo 1 chiều nhất định.

Băng tải có kích thước rộng 6cm, dài 50cm, kích thước con lăn có đường kính 2,8cm và động cơ DC quay với tốc độ khoảng 280 vòng/phút. Ta tính được vận tốc của băng tải như sau:

V = C x n = D x 3,14 x n = 2,8 x 10-2 x 3,14 x 280 = 24,6 (mét/phút) (3.3)

Trong đó:

V: vận tốc của băng tải

C: chu vi con lăn

n: tốc độ quay động cơ

D: đường kính con lăn

### Khối phân loại

Khối phân loại sử dụng Servo để gạc sản phẩm vào vị trí chứa. Ta cấp nguồn 5VDC để servo hoạt động, ở chân xung cấp 1 xung từ 1ms đến 2ms để điều khiển góc quay theo ý muốn.



Hình 3. Chu kì hoạt động của động cơ servo

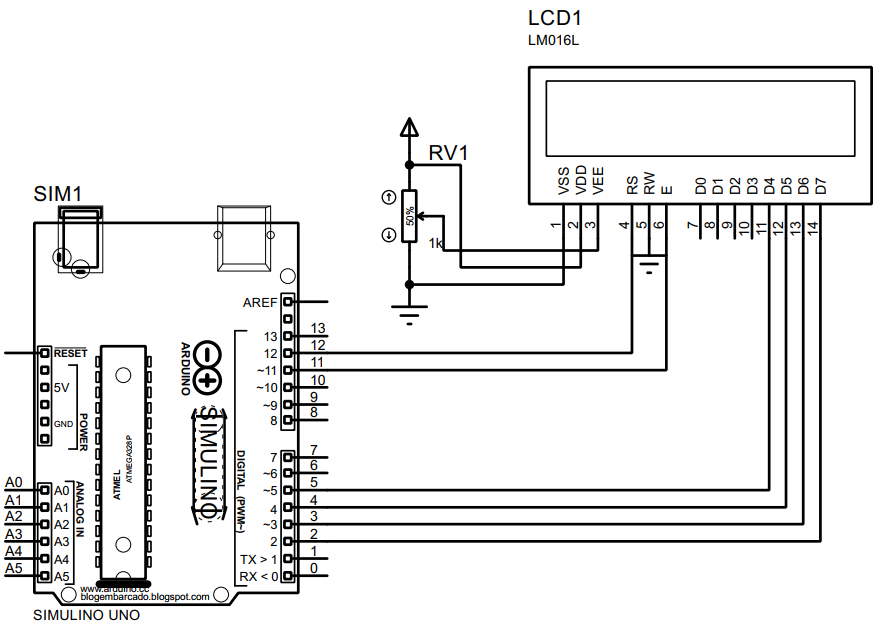
Với độ rộng xung từ 1ms (1000us) => 2ms (2000us) thì servo được đặt với góc lệch tương ứng trong khoảng 0 =>180 (độ). Để gạt sản phẩm vào vị trí chứa mong muốn ta sẽ điều khiển Servo quay 1 góc khoảng 50 độ tương ứng với độ rộng xung khoảng 1.25ms.

### Khối hiển thị

Khối hiển thị sử dụng LCD 16x2 để hiển thị giá trị đếm được của các sản phẩm phân loại.

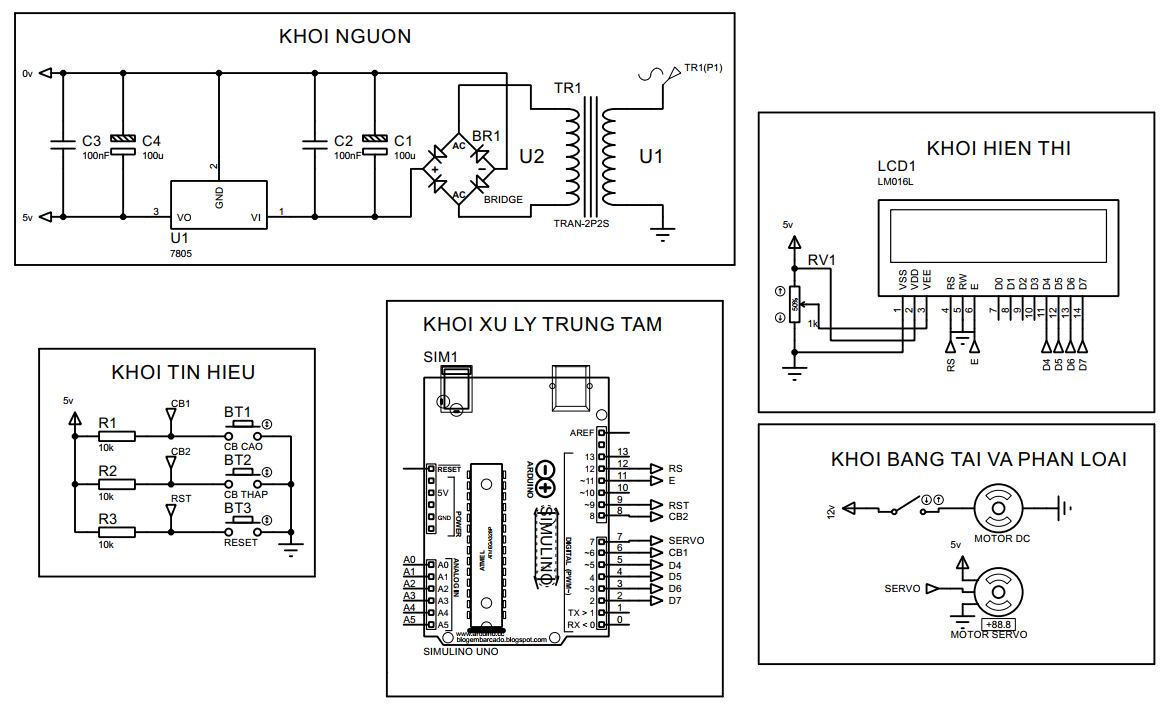
LCD có thể được giao tiếp với vi điều khiển ở chế độ 8 bit hoặc ở chế độ 4 bit. Chế độ 8 bit sử dụng 8 bus dữ liệu và chân RS, R/W, E để hoạt động. Tuy nhiên chế độ 4 bit chỉ sử dụng 4 bus dữ liệu cùng với các chân điều khiển. Điều này sẽ giúp tiết kiệm số lượng chân I/O nhằm phục vụ cho mục đích khác. Vì vậy trong đề tài này ta sẽ sử dụng phương thức giao tiếp 4 bit.

Trong chế độ 4 bit, chỉ có 4 đường D4-D7, cùng với các chân RS, R/W và E được sử dụng. Việc này giúp chúng ta tiết kiệm được 4 chân của vi điều khiển để sử dụng cho những mục đích khác. Ở đây chúng ta chỉ cần ghi vào mô-đun LCD. Vì vậy, chân R/W được nối đất. Các byte dữ liệu được chia thành 24 bit và được chuyển dưới dạng một nibble. Việc truyền dữ liệu tới LCD được thực hiện bằng cách gán trạng thái logic cho các chân điều khiển RS và E.



Hình 3. Sơ đồ nối chân LCD với Ardunio

### 3.3.7. Sơ đồ nguyên lý toàn mạch



Hình 3. Sơ đồ nguyên lý toàn mạch

### 3.3.8. Nguyên lý hoạt động

Khi được cấp nguồn, mạch sẽ hoạt động theo chương trình được thiết lập trên Arduino. Chu trình hoạt động của mô hình hệ thống này được bắt đầu từ sự thu nhận tín hiệu của các cảm biến. Các tín hiệu được trung tâm xử lý và xuất các lệnh tương thích cho Servo cùng với LCD để thực hiện theo đúng yêu cầu đề tài.

Ở trạng thái bình trường, cảm biến E18-D80NK cấp xung mức cao cho bộ xử lý trung tâm. Với xung mức cao, thông qua code đã được lập trình, Arduino Uno R3 duy trì hiện trạng. Động cơ Servo giữ nguyên thanh gạc phân loại ở góc 0 độ.

Khi được tác động bởi đối tượng, dòng điện qua trở treo thông qua cảm biến E18-D80NK tới GND. Lúc này, xung mức thấp được cấp cho Arduino Uno R3. Dữ liệu được xử lý và đưa vào LCD làm thay đổi sự hiển thị của LCD. Đồng thời, động cơ Servo quay thanh gạc phân loại một góc 50 độ, đối tượng được đưa vào nơi chứa tương ứng.

Khi nhấn nút Reset, xung mức thấp được cấp vào chân số 9 của Arduino Uno R3. Tất cả dữ liệu đếm được sẽ được reset về giá trị 0 và hiển thị lên LCD.

## 3.4 Thi Công Hệ Thống

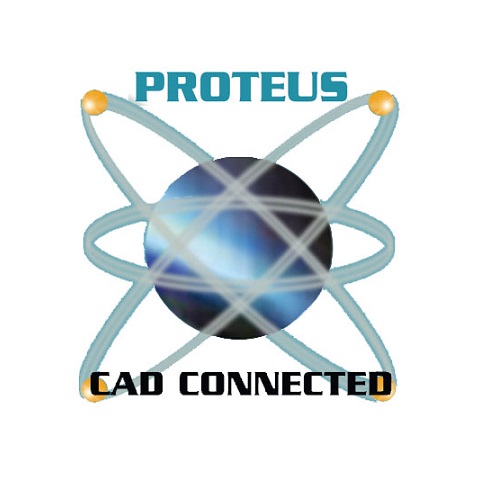
### 3.4.1 Thi công mạch

#### Mạch nguyên lý

##### a. Giới thiệu phần mềm Proteus

Phần mềm Proteusđược biết đến là phần mềm vẽ, mô phỏng mạch điện tử, được phát triển bởi công ty Lacenter Electronics. Với proteus, người dùng có thể mô tả hầu hết được những linh kiện điện tử thông minh phổ biến hiện nay. Ngoài ra, nó còn hỗ trợ cho những phần mềm khác như 8051, PIC, Motorola, AVR.

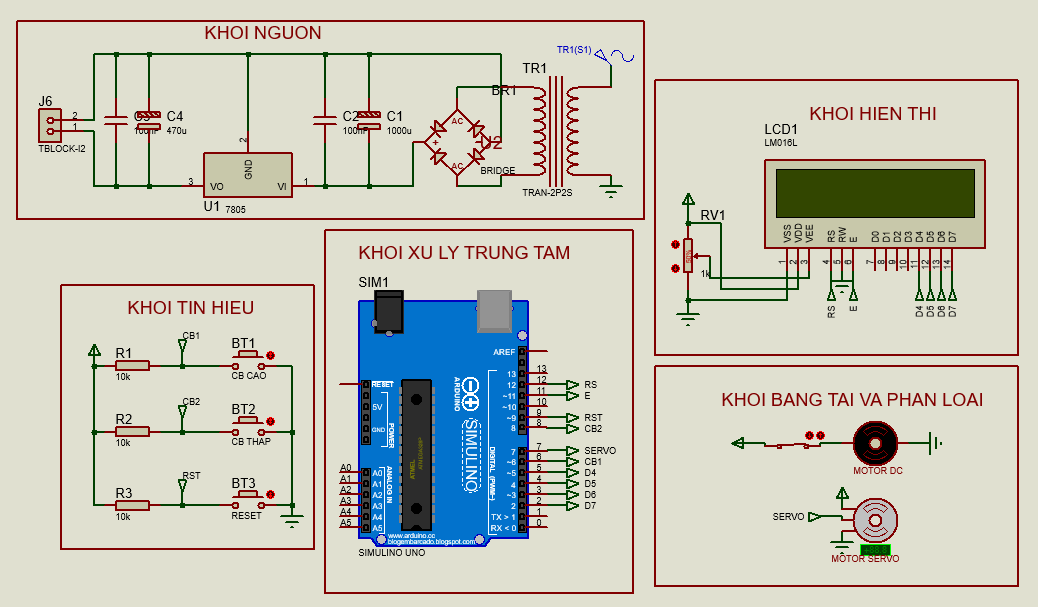
Tính năng được yêu thích nhất của Proteus chính là có thể mô phỏng hoạt động của những mạch điện tử, cụ thể là thiết kế mạch và viết trình điều khiển cho các loại vi điều khiển như MCS – 51, AVR, PIC, …



Hình 3. Phần mềm proteus

##### b. Các bước vẽ mạch nguyên lý bằng phần mềm Proteus

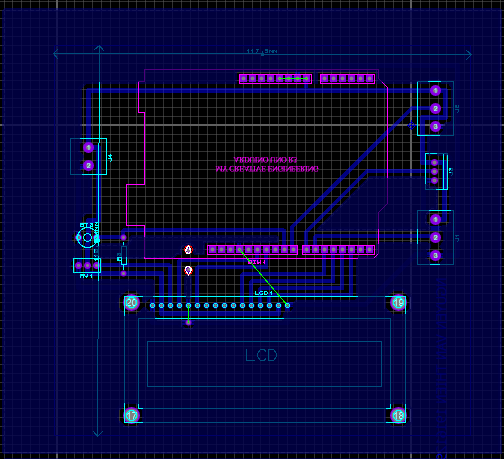
* Bước 1: Khởi động phần mềm và tạo Project mới.
* Bước 2: Mở chương trình ISIS Professional.
* Bước 3: Lấy tất cả các linh kiện cần sử dụng từ thư viện.
* Bước 4: Tiến hành bố trí linh kiện.
* Bước 5: Nối dây để kết nối linh kiện.



Hình 3. Sơ đồ mạch nguyên lý

### 3.4.2 Mạch in PCB

Yếu tố được xem là quang trọng nhất khi vẽ mạch PCB đó là kích thước đường dây nguồn và dây tín hiệu của mạch. Kích thước đường dây sẽ ảnh hưởng đến việc thi công mạch và hàn linh kiện. Để dễ dàng cho việc thi công mạch thì em đi dây nguồn là T60 và dây tín hiệu là T50.



Hình 3. Mạch in PCB

## 3.5 Thi công mạch điều khiển

### 3.5.1 Phần mềm lập trình Ardunio

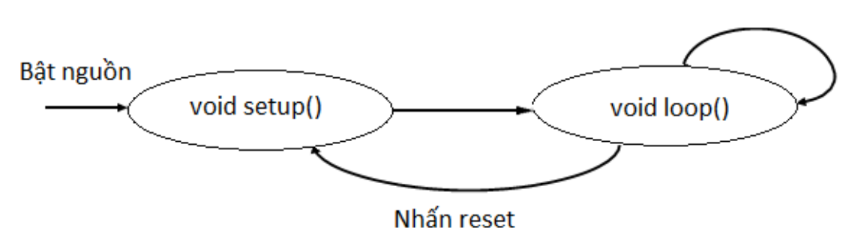
Môi trường phát triển tích hợp Arduino IDE là một ứng dụng đa nền tảng được viết bằng Java, và được dẫn xuất từ IDE cho ngôn ngữ lập trình xử lý và các dự án lắp ráp. Do có tính chất mã nguồn mở nên môi trường lập trình này hoàn toàn miễn phí và có thể mở rộng thêm bởi người dùng có kinh nghiệm.

Người sử dụng chỉ cần định nghĩa hai hàm để thực hiện một chương trình hoạt

động theo chu trình:

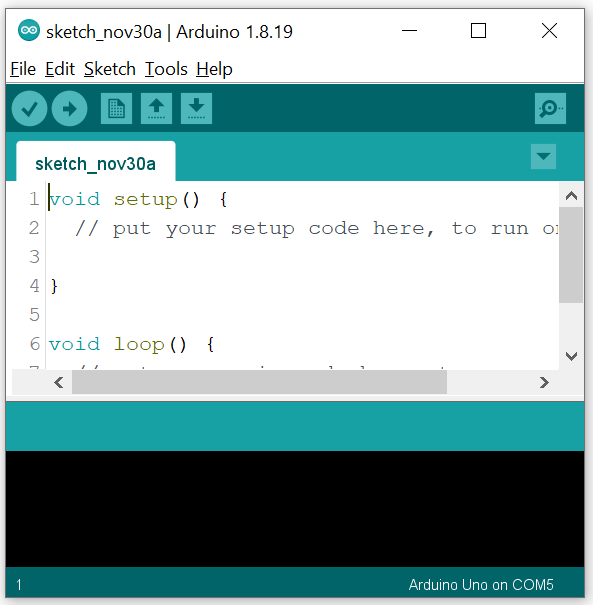
* + Setup(): hàm chạy một lần duy nhất vào lúc bắt đầu của một chương trình dùng để khởi tạo các thiết lập.
  + Loop(): hàm được gọi lặp lại liên tục cho đến khi bo mạch được tắt.

Chu trình đó có thể mô tả trong hình dưới đây:



Hình 3. Quy trình làm việc của ardunio

Arduino IDE là nơi để soạn thảo code, kiểm ra lỗi và upload code.



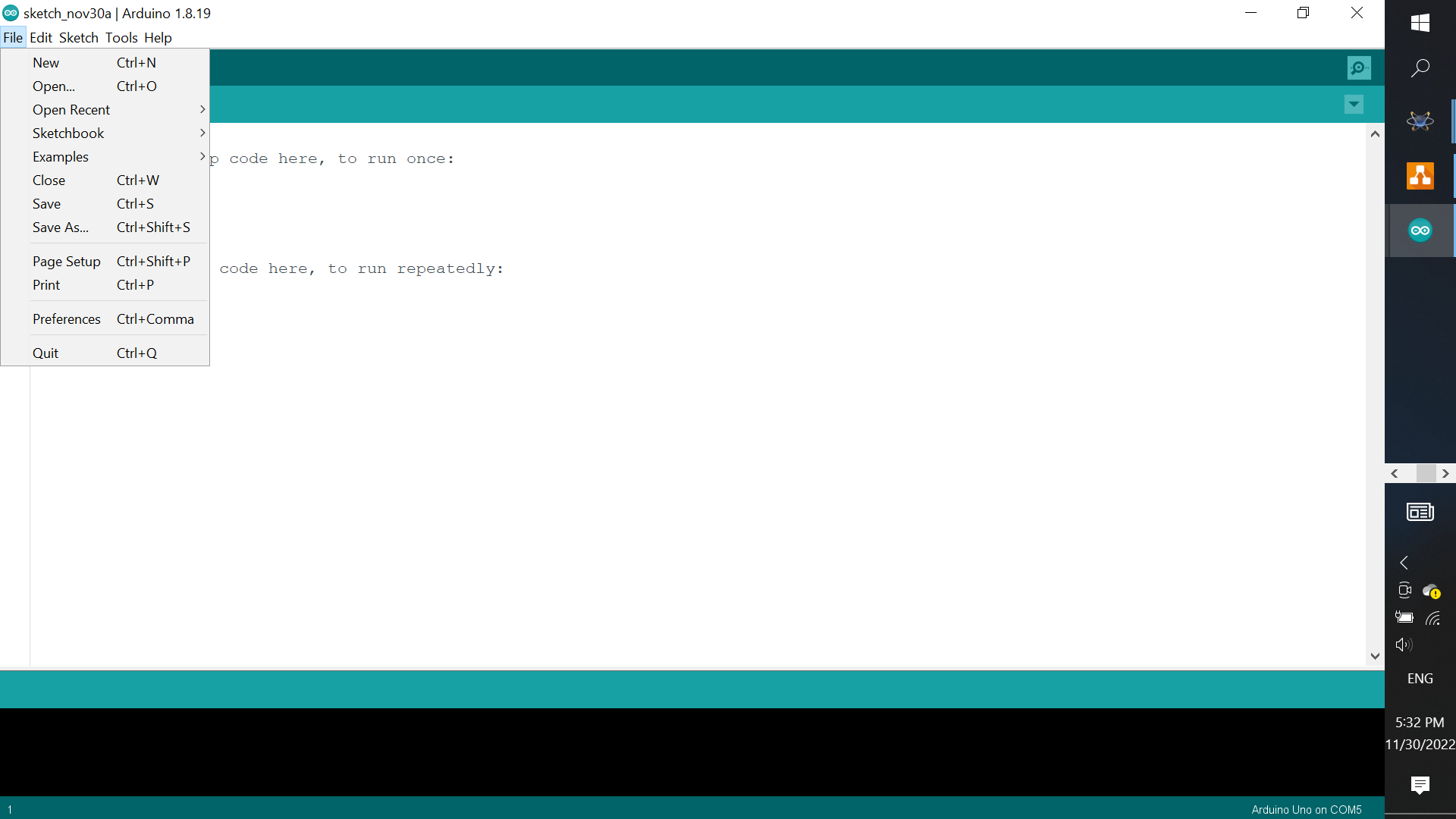
Hình 3. Giao diện lập trình của Ardunio

Arduino IDE Menu:



Hình 3. Giao diện menu Ardunio IDE

Filemenu:



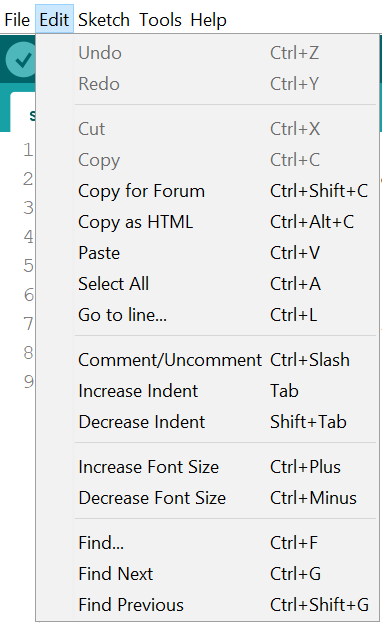
Hình 3. Giao diện file menu Ardunio IDE

Trong file menu chúng ta cần quan tâm tới mục Examples, vì đây là nơi chứa các

code chương trình mẫu như điều khiển nhấp nháy led (Blink), cách sử dụng các chân

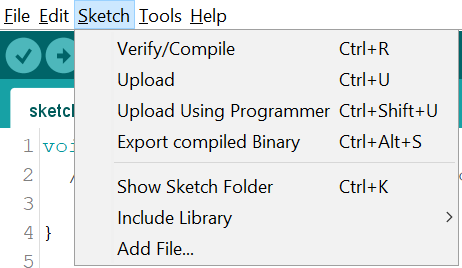
digital, analog, cảm biến,...

Edit menu:



Hình 3. Giao diện edit menu Ardunio IDE

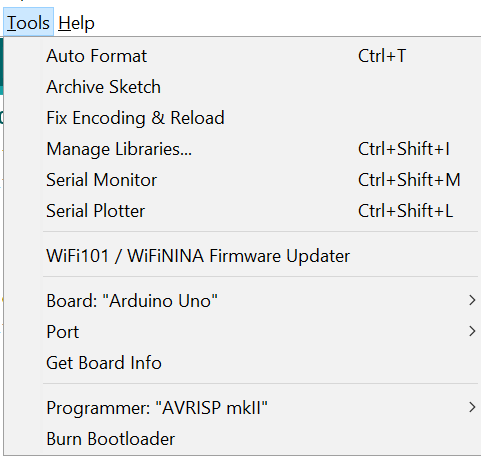
Sketch menu:



Hình 3. Giao diện sketch menu Ardunio IDE

* Verify/ Compile: chức năng kiểm tra lỗi code.
* Show Sketch Folder: hiển thị nơi code được lưu.
* Add File: thêm vào một Tap code mới.
* Import Library: thêm thư viện từ bên ngoài cho IDE.

Tool menu:



Hình 3. Giao diện tools menu Ardunio IDE

Trong mục Tools ta quan tâm các mục Board và Serial Port. Trong mục Board,

cần phải lựa chọn board mạch cho phù hợp với loại board sử dụng. Nếu sử dụng loại

board khác thì phải chọn đúng loại board mà mình đang có, nếu sai thì khi Upload

chương trình vào chip sẽ bị lỗi. Vì sử dụng Arduino Uno R3 cho đề tài này

nên chúng ta sẽ chọn Board: “Arduino Uno ”.

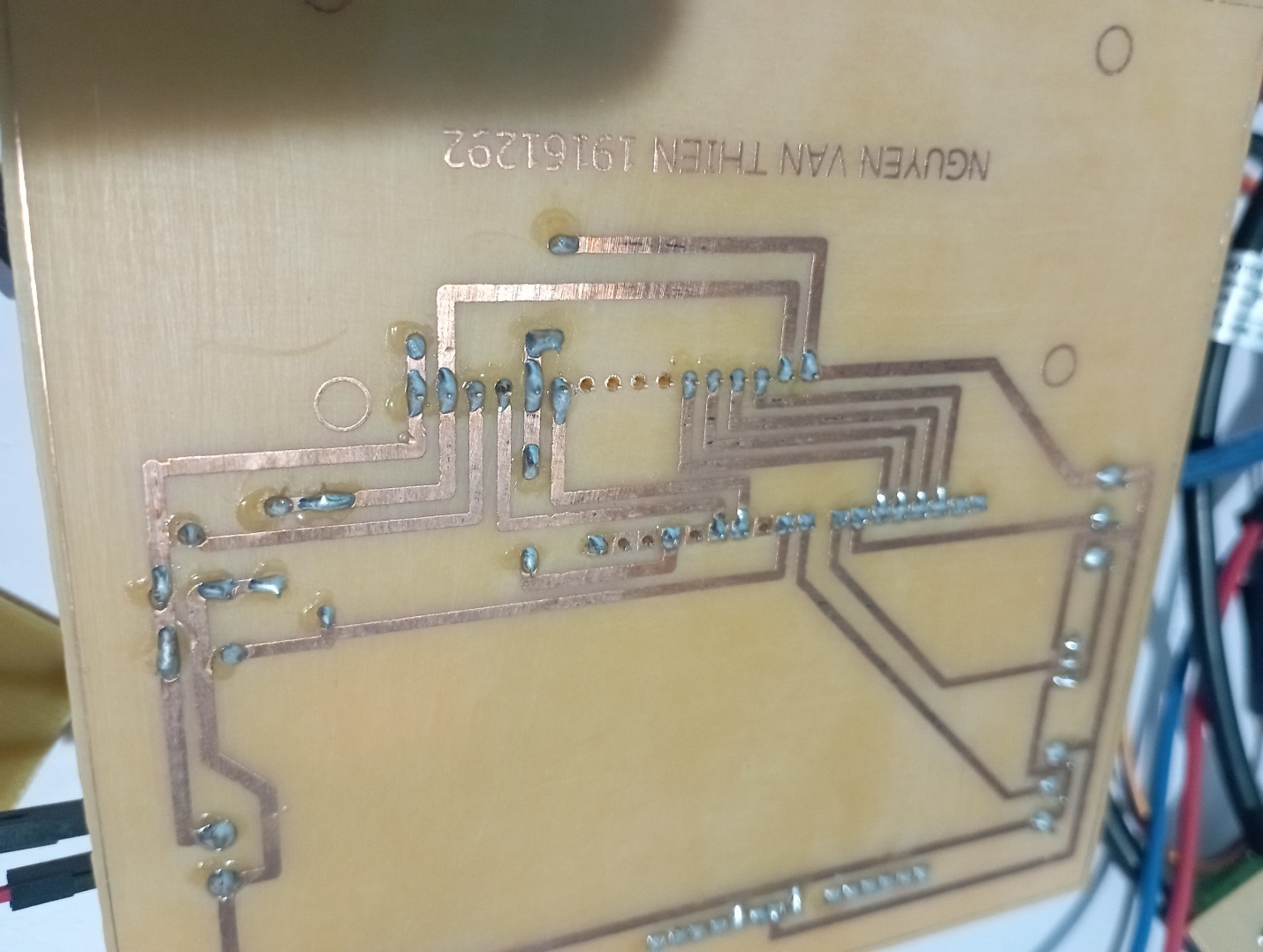
Serial Port: đây là nơi lựa chọn cổng Com của Arduino. Khi chúng ta cài đặt driver thì máy tính sẽ hiện thông báo tên cổng Com của Arduino là bao nhiêu, ta chỉ việc vào Serial Port chọn đúng cổng Com để nạp code, nếu chọn sai thì không thể nạp code cho Arduino được.

Arduino Toolbar có một số button và chức năng của chúng như sau:

* Verify: kiểm tra code có lỗi hay không.
* Upload: nạp code đang soạn thảo vào Arduino.
* New, Open, Save: Tạo mới, mở và lưu sketch.

### 3.5.2 Thi công mạch

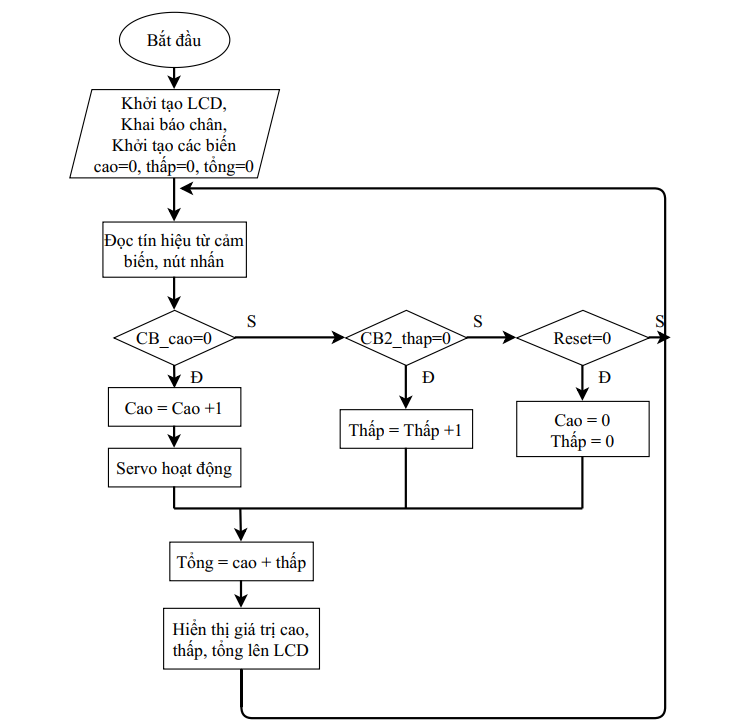
Sau khi đã hoàn tất mạch in trên Proteus, hàn mạch ta được mạch hoàn thiện sau đây:



Hình 3. Mạch sau khi in thành công

## 3.6. Lưu đồ giải thuật hệ thống

### 3.6.1 Lưu đồ giải thuật



Hình 3. Lưu đồ giải thuật hệ thống

* Giải thích lưu đồ:

Chương trình bắt đầu vào khởi tạo LCD, khai báo chân, khởi tạo các biến. Vòng lặp chương trình được thực hiện. Bắt đầu vòng lặp bằng việc đọc tín hiệu từ các cảm biến và nút nhấn. Tiếp theo sẽ kiểm tra xem cảm biến cao có tín hiệu hay không, nếu đúng thì tăng giá trị cao lên 1 đơn vị đồng thời điều khiển servo quay góc 50 độ, nếu sai thì kiểm tra xem cảm biến thấp có tín hiệu hay không, nếu đúng thì tăng giá trị thấp lên 1 đơn vị, nếu sai thì kiểm tra xem có nhấn reset hay không, nếu đúng thì set giá trị cao và thấp đều bằng 0, nếu sai thì quay lại đầu vòng lặp. Tính giá trị tổng bằng tổng của giá trị cao và giá trị thấp. sau đó hiển thị tất cả các giá trị cao, thấp, tổng lên LCD và quay lại vòng lặp.

## 3.7 Chương trình điều khiển

#include <LiquidCrystal.h>

#include <Servo.h>

LiquidCrystal LCD(12,11,5,4,3,2);

Servo servo;

int sensorPin1=6;

int sensorPin2=8;

int resetPin=9;

int cao=0,thap=0,tong=0;

int sensor1,sensor2,reset;

void setup() {

LCD.begin(16,2);

LCD.print("SP:cao thap tong");

LCD.setCursor(0,1);

LCD.print("SL:");

pinMode(sensorPin1,INPUT);

pinMode(sensorPin2,INPUT);

pinMode(resetPin,INPUT);

servo.attach(7);

servo.write(0);

}

void loop() {

reset=digitalRead(9);

sensor1=digitalRead(6);

sensor2=digitalRead(8);

if(sensor1==0){

cao=cao+1;

delay(100);

servo.write(50);

delay(4000);

servo.write(0);

}

if(sensor2==0){

thap=thap+1;

delay(1000);

}

if(reset==0){

cao=0;

thap=0;

tong=0;

LCD.setCursor(5,1);

LCD.print(" ");

LCD.setCursor(9,1);

LCD.print(" ");

LCD.setCursor(14,1);

LCD.print(" ");

}

tong = cao + thap;

LCD.setCursor(4,1);

LCD.print(cao);

LCD.setCursor(8,1);

LCD.print(thap);

LCD.setCursor(13,1);

LCD.print(tong);

}

# CHƯƠNG 4: KẾT QUẢ\_KẾT LUẬN\_ĐÁNH GIÁ

## 4.1. Kết quả

### 4.1.1. Kết quả thực hiện

Qua thời gian 15 tuần nghiên cứu, nổ lực học tâp, tìm hiểu dưới sự hướng dẫn của Thầy Nguyễn Đình Phú, đồ án môn học này em đã hoàn thành đúng thời hạn quy định, giải quyết các yêu cầu đặt ra ban đầu là thiết kế và thi công hệ thống đếm và phân loại sản phẩm theo chiều cao. Nghiên cứu tìm hiểu về cách hoạt động và cách kết nối của các linh kiện có trong mạch. Thực hiện đề tài em đã biết sử dụng các phần mềm mô phỏng, thiết kế mạch như protues, phần mềm Arduino IED để viết chương trình cho Arduino hoạt động... Biết cách sử dụng, chức năng, nguyên lý hoạt động của Arduino Uno, cảm biến hồng ngoại E18-D80NK, motor Servo,… Ngoài ra, còn có thể cải thiện tốt hơn cách hàn mạch, thi công mạch in.

Kết quả thực hiện: Sắp xếp linh kiện hợp lý có tính thẩm mỹ cao. Mạch hoạt động tương đối ổn định, khá chính xác, giá cả phù hợp và có thể dễ dàng ứng dụng vào thực tế, sau khi hoàn thành đã đạt khoảng 90% như yêu cầu, đạt đầy đủ các chức năng cơ bản như:

* Tìm hiểu về Arduino Uno.
* Tìm hiểu về cảm biến hồng ngoại E18-D80NK, motor Servo.
* Tìm hiểu về cách hoạt động và cách kết nối của các linh kiện có trong hệ thống.

### 4.1.2. Sản phẩm thực tế



Hình 4. Sản phẩm hoàn thiện

## 4.2 Kết luận

Sau một thời gian dài nghiên cứu, xây dựng ý tưởng và bắt tay vào việc tính toán,

thi công. Em đã hoàn thành đề tài “Thiết kế và thi công hệ thống đếm và phân loại sản phẩm theo chiều cao”. Mô hình đã đạt được độ ổn định tương đối về mặt vận hành cũng như điều khiển.

Quá trình nghiên cứu và xây dựng đề tài giúp em hiểu được nhiều thiết bị ứng dụng trong ngành tự động hóa hiện nay nhằm nâng cao chất lượng cuộc sống của con người.

Tuy nhiên, đây chỉ mới là mô hình nhỏ đồng thời do sự hạn chế về mặt kiến thức cũng như về thời gian nên việc tính toán thiết kế chưa có đi sâu được, chính vì vậy việc thi công mô hình vẫn chưa đạt được chuẩn mực nhất định. Trên cơ cở của mô hình này, nếu như muốn phát triển mô hình này để đưa vào sản xuất công nghiệp thì việc tính toán phải hết sức kĩ lưỡng đến từng chi tiết nhỏ. Tùy theo chức năng và yêu cầu của hệ thống mà ta tính toán cho phù hợp.

## 4.3 Hướng phát triển của đề tài

* Áp dụng cho các hệ thống dây chuyền sản xuất nhỏ.
* Thay đổi cảm biến để tạo ra dây chuyền phân loại dựa trên các tiêu chí khác nhau của sản phẩm.
* Phát triển thêm giám sát trên máy tính.
* Thiết kế thêm các hệ thống phân biệt màu, vật liệu cho hệ thống đa dạng.

# TÀI LIỆU THAM KHẢO

[1] Nguyễn Đình Phú, Nguyễn Trường Duy, Nguyễn Duy Thảo, Hà A Thồi, Giáo Trình Thực Hành: Kỹ Thuật Số, khoa Điện-Điện Tử, trường ĐHSPKT, Tp.HCM, 2016.

[2] Nguyễn Đình Phú, Nguyễn Trường Duy, Giáo Trình: Kỹ Thuật Số, Xuất bản ĐH Quốc Gia, Tp.HCM, 2013.

[3] https://docs.arduino.cc/resources/datasheets/A000066-datasheet.pdf

[4] Arduino [online]: <http://arduino.vn/bai-viet/42-arduino-uno-r3-la-gi>

[5] <https://hshop.vn/products/cam-bien-vat-can-hong-ngoai-e18-d80nk->4

[6] <https://dientu360.com/dong-co-servo-sg90>

[7] [www.sparkfun.com](http://www.sparkfun.com), “Datasheets LCD ADM1602K-NSW-FBS-3.3v”

[8] https://datasheetspdf.com/pdf/791970/TowerPro/SG90/1