**TRƯỜNG ĐẠI HỌC VINH**

**VIỆN KĨ THUẬT VÀ CÔNG NGHỆ**



BÀI BÁO CÁO ĐỒ ÁN

HỌC PHẦN KỸ THUẬT LẬP TRÌNH

NGÀNH KỸ THUẬT ĐIỀU KHIỂN VÀ TỰ ĐỘNG HÓA

**TÊN ĐỀ TÀI :** **Đếm số lượng sản phẩm sử dụng cảm biến hồng ngoại, hiển thị lên màn hình LCD thông qua Arduino**

**GV hướng dẫn : Thầy MAI THẾ ANH**

**Sinh viên thực hiện: HỒ ĐÌNH HUÂN 235752021610049**

**TRẦN DANH HOÀNG 235752021610062**

**TRỊNH QUANG HUY 235752021610016**

**LÔ QUỐC HỘI 235752021610045**

**NGUYỄN TRỌNG HỒNG 235752021610046**

**ĐẬU THẾ HOÀNG 235752021610017**

**Khóa : 64**

**NGHỆ AN – 2024**

**TRƯỜNG ĐẠI HỌC VINH CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM VIỆN KỸ THUẬT VÀ CÔNG NGHỆ Độc Lập - Tự Do - Hạnh Phúc**

**MỤC TIÊU VÀ NHIỆM VỤ**

**Họ và tên**: - Hồ Đình Huân MSSV: 235752021610049

- Trần Danh Hoàng MSSV: 235752021610062

- Đậu Thế Hoàng MSSV: 235752021610017

- Trịnh Quang Huy MSSV: 235752021610016

- Nguyễn Trọng Hồng MSSV: 235752021610046

- Lô Quốc Hội MSSV: 235752021610045

**Ngành**: Kỹ thuật điều khiển và tự động hóa................................................

**1. Mục tiêu đồ án** (tên đồ án):

Đếm số lượng sản phẩm sử dụng cảm biến hồng ngoại, hiển thị lên màn hình LCD thông qua Arduino…………………………………………………………………

**2. Nhiệm vụ**:

Thiết kế mô hình đếm số lượng sản phẩm sử dụng cảm biến hồng ngoại, hiển thị lên màn hình LCD thông qua Arduino. Thiết kế 15-20 trang slide để thuyết trình. Hoàn thành bài báo cáo để nạp cho giám thị chấm thi. Hoàn hiện các mục tiêu trên đúng hạn.

**3. Ngày giao đồ án**  tháng năm 2024

**4. Ngày hoàn thành đồ án** tháng năm 2024

**5. Người hướng dẫn:** Thầy MAI THẾ ANH

Nghệ An, ngày 4 tháng 1 năm 2025

**NGƯỜI HƯỚNG DẪN**

**MAI THẾ ANH**

**TRƯỜNG ĐẠI HỌC VINH CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM VIỆN KỸ THUẬT VÀ CÔNG NGHỆ Độc Lập - Tự Do - Hạnh Phúc**

**PHÂN CÔNG NHIỆM VỤ ĐỒ ÁN**

**Tên đồ án:**.…. Đếm số lượng sản phẩm sử dụng cảm biến hồng ngoại, hiển thị lên màn hình LCD thông qua Arduino.

**Ngành:** ……………..Kỹ thuật điều khiển và tự động hóa ....................................

**GV hướng dẫn:** …Thầy Mai Thế Anh ...................................................

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Họ và tên | Nhiệm vụ | Tự đánh giá | Chữ ký |
| 1 | HỒ ĐÌNH HUÂN  235752021610049 | * Lên kế hoạch, ý tưởng nội dung cần làm, phân công nhiệm vụ cho từng thành viên * Tổng hợp các ý kiến của từng thành viên trong quá trình làm * Lập trình đếm sản phẩm hiển thị lên màn hình LCD * Làm báo cáo | A |  |
| 2 | TRẦN DANH HOÀNG  235752021610064 | * Lên ý tưởng, thiết kế đồ án * Lập trình đếm sản phẩm hiển thị lên màn hình LCD * Làm báo cáo | A |  |
| 3 | TRỊNH QUANG HUY  235752021610016 | * Lập trình đếm sản phẩm hiển thị lên màn hình LCD * Làm báo cáo và chỉnh sửa | B+ |  |
| 4 | LÔ QUỐC HỘI  235752021610045 | * Lập trình đếm sản phẩm hiển thị lên màn hình LCD * Thiết kế slide | B |  |
| 5 | NGUYỄN TRỌNG HỒNG  235752021610046 | * Lập trình đếm sản phẩm hiển thị lên màn hình LCD * Thuyết trình | B+ |  |
| 6 | ĐẬU THẾ HOÀNG  235752021610017 | * Lập trình đếm sản phẩm hiển thị lên màn hình LCD * Thiết kế slide | B+ |  |

Nghệ An, ngày.4..tháng.1..năm 2025

**Nhóm trưởng**

**HỒ ĐÌNH HUÂN**

Mục Lục

[CHƯƠNG 1. TỔNG QUAN VỀ CÁC THIẾT BỊ ĐƯỢC SỬ DỤNG 1](#_Toc186819121)

[**1.1. Khái quát về Arduino** 1](#_Toc186819122)

[1.1.1. Lịch sử phát triển của Arduino 1](#_Toc186819123)

[1.1.2. Một số đặc điểm,cấu tạo của Arduino 2](#_Toc186819124)

[1.1.3. Một số ứng dụng của Arduino 4](#_Toc186819125)

[CHƯƠNG 2. THIẾT KẾ PHẦN CỨNG VÀ LẬP TRÌNH TRÊN ARDUINO IDE 6](#_Toc186819126)

[**2.1. Sơ đồ khối** 6](#_Toc186819127)

[**2.2. Chức năng của các khối** 6](#_Toc186819128)

[2.2.1. Khối cảm biến 6](#_Toc186819129)

[2.2.2. Khối hiển thị 7](#_Toc186819130)

[2.2.3. Khối vi điều khiển và khối nguồn 9](#_Toc186819131)

[**2.3. Nguyên lý hoạt động** 10](#_Toc186819132)

[**2.4. Phần mềm Arduino IDE** 10](#_Toc186819133)

[2.4.1. Giới thiệu phần mềm Arduino IDE 10](#_Toc186819134)

[2.4.2. Lập trình trên Arduino IDE 12](#_Toc186819135)

[TÀI LIỆU THAM KHẢO 20](#_Toc186819136)

**LỜI NÓI ĐẦU**

Trong thời đại số hiện nay, kỹ thuật lập trình đóng vai trò vô cùng quan trọng. Nó không chỉ là một kỹ năng chuyên môn mà còn là một công cụ mạnh mẽ để tạo ra những thay đổi tích cực.Lập trình là nền tảng để tạo ra các ứng dụng, phần mềm, nền tảng mới, thúc đẩy sự đổi mới và sáng tạo trong mọi lĩnh vực. Từ các ứng dụng di động đơn giản đến các hệ thống trí tuệ nhân tạo phức tạp, tất cả đều được xây dựng dựa trên mã code.Lập trình giúp tự động hóa các công việc lặp đi lặp lại, giảm thiểu lỗi sai và tăng năng suất làm việc. Các doanh nghiệp sử dụng phần mềm để quản lý quy trình, phân tích dữ liệu, và đưa ra quyết định kinh doanh hiệu quả hơn.

Arduino là một nền tảng điện tử mã nguồn mở, được thiết kế để giúp mọi người dễ dàng tạo ra các dự án tương tác và hỗ trợ cho con người rất nhiều trong lập trình và thiết kế. Nó bao gồm cả phần cứng (các bo mạch Arduino) và phần mềm (môi trường lập trình Arduino IDE). Arduino đã trở nên rất phổ biến trong cộng đồng người đam mê điện tử, giáo dục, và cả trong các ứng dụng công nghiệp nhỏ.

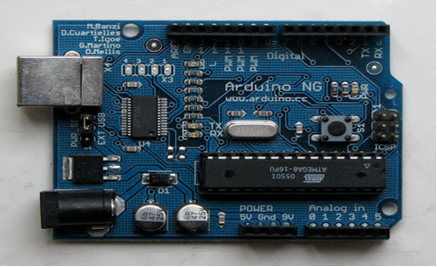
Trên cơ sở kiến thức đã học trong học phần Kỹ thuật lập trình cùng với những hiểu biết, nhóm em đã quyết định thực hiện đề tài: “Đếm số lượng sản phẩm sử dụng cảm biến hồng ngoại, hiển thị lên màn hình LCD thông qua Arduino”. Nhóm em cũng xin cảm ơn thầy Mai Thế Anh đã tận tình chỉ dạy, hướng dẫn nhóm em hoàn thành được sản phẩm của mình !

# CHƯƠNG 1. TỔNG QUAN VỀ CÁC THIẾT BỊ ĐƯỢC SỬ DỤNG

## 1.1. Khái quát về Arduino

### 1.1.1. Lịch sử phát triển của Arduino

Arduino được khởi động vào năm 2005 như là một dự án dành cho sinh viên trại Interaction Design Institute Ivrea tại Italy. Cái tên "Arduino" đến từ một quán bar tại Ivrea, nơi một vài nhà sáng lập của dự án này thường xuyên gặp mặt. Trên hình 1 là Arduino những năm đầu.



Hình 1.1: Arduino UNO ( 2006 )

Lý thuyết phần cứng được đóng góp bởi một sinh viên người Colombia tên là Hernando Barragan. Sau khi nền tảng Wiring hoàn thành, các nhà nghiên cứu đã làm việc với nhau để giúp nó nhẹ hơn, rẻ hơn, và khả dụng đối với cộng đồng mã nguồn mở. Giá hiện tại của board mạch này dao động xung quanh $30 và được làm giả đến mức chỉ còn $9. Một mạch bắt chước đơn giản Arduino Mini Pro có lẽ được xuất phát từ Trung Quốc có giá rẻ hơn $4. Arduino phổ biến nhất hiện nay là Arduino UNO R3.

Dù hầu như không có một sự tiếp thị hay quảng cáo nào nhưng tin tức về Arduino vẫn lan truyền với tốc độ chóng mặt nhờ vô vàn lời truyền miệng tốt đẹp của những người dùng đầu tiên.

Theo ước tính, khoảng giữa năm 2011 đã có hơn 300 ngàn mạch Arduino chính thức đã được sản xuất thương mại. Đến năm năm 2013 có khoảng 700 ngàn mạch chính thức đã được đưa tới tay người sử dụng.

Hiện nay Arduino nổi tiếng trên toàn thế giới đến nỗi có người đã tìm đến thị trấn Ivrea chỉ để tham quan nơi đã sản sinh ra nền tảng thú vị này. Nó cũng được sử dụng rộng rãi trong lập trình,thiết kế những bài toán nhỏ,dễ làm dành cho sinh viên học tập, nghiên cứu, làm đồ án...với giá thành rất rẻ, dễ mua, dễ làm, thuận tiện trong công việc.

### 1.1.2. Một số đặc điểm,cấu tạo của Arduino

**a. Đặc điểm**

Arduino là nền tảng tạo mẫu điện tử mã nguồn mở, được sử dụng nhằm xây dựng các ứng dụng điện tử tương tác với nhau hoặc với môi trường được thuận tiện, dễ dàng hơn.

Arduino là công cụ hỗ trợ đắc lực cho công việc lập trình. Điểm hấp dẫn ở Arduino với người đam mê lập trình là ngôn ngữ dễ học khá giống C/C++, các ngoại vi trên bo mạch đều đã được chuẩn hóa nên không cần biết nhiều về điện tử, chúng ta cũng có thể lập trình được những ứng dụng thú vị. Thêm nữa Arduino là một platform đã được chuẩn hóa nên đã có rất nhiều các bo mạch mở rộng (shield) để cắm chồng lên bo mạch Arduino, có thể hình dung dễ hiểu là "library" của các ngôn ngữ lập trình.

Nền tảng mẫu này giống như một máy tính thu nhỏ, giúp người dùng lập trình và thực hiện các dự án điện tử mà không cần phải đến các công cụ chuyên dụng để phục vụ việc nạp code.

Phần mềm này tương tác với thế giới bên ngoài thông qua các cảm biến điện tử, đèn và động cơ.

**b. Cấu tạo**

Phần cứng Arduino là bảng mạch nguồn mở, cùng bộ vi xử lý và chân đầu vào/ đầu ra (I/ O) để liên lạc, điều khiển các đối tượng vật lý (LED, servo, nút, v.v.). Bảng mạch thường được cấp nguồn qua USB hoặc nguồn điện bên ngoài, cho phép cung cấp năng lượng cho các phần cứng, cảm biến khác.

Là phần mềm nguồn mở tương tự như C ++. Môi trường phát triển tích hợp Arduino (IDE – Integrated Development Environment) cho phép bạn soạn thảo, biên dịch code, nạp chương cho board.

⇒ Tất cả những điều này nhằm hỗ trợ cho các sinh viên, nhà sản xuất tự do phát triển ý tưởng của họ thành các đối tượng thực sự một cách dễ dàng.

Chi tiết phần cứng của Arduino:

1. Cổng USB: là chân cắm để tải mã lập trình từ PC lên chip điều khiển. Đồng thời đây cũng là cổng giao tiếp serial giúp truyền dữ liệu từ chip điều khiển vào máy tính.

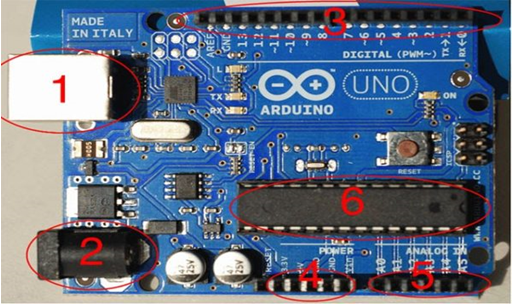
2. Jack nguồn: để chạy Arduino, bạn hoàn toàn có thể nạp nguồn từ cổng USB ở trên. Tuy nhiên không phải lúc nào cũng kết nối với máy tính được. Có những dự án cần thực hiện ngoài trời sẽ cần một nguồn điện khác với mức điện áp từ 9V -12V.

3. Hàng Header: những chân đánh số từ 0 – 12 là hàng digital pin. Đây là nơi truyền – nhận các tín hiệu số. Bên cạnh đó sẽ có một chân nối đất (GND) và pin điện áp tham chiếu (AREF).

4. Hàng header thứ 2: chủ yếu liên quan tới điện áp đất, nguồn.

5. Hàng header thứ 3: đây là các chân để nhập – xuất các tín hiệu analog (đọc thông tin của các thiết bị cảm biến).

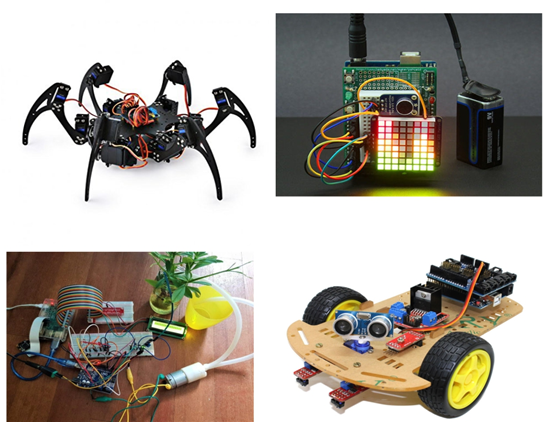
6. Chip điều khiển AVR: bộ phận xử lý trung tâm của toàn bo mạch. Với mỗi mẫu Arduino khác nhau, con chip này sẽ khác nhau. Ví dụ trên Arduino Uno thì sẽ sử dụng ATMega328.



Hình 1.2: Cấu tạo của Arduino

### 1.1.3. Một số ứng dụng của Arduino

Với sự phát triển nhanh chóng của mã nguồn mở Arduino việc học đã không còn là khó khăn cho những ai đam mê về điện tử, lập trình. Vì vậy, hiện nay Arduino đã được các trường THCS, THPT, Cao Đẳng, Đại Học đưa vào giảng dạy và làm các đề tài và hoạt động ngoại khóa của nhà trường. Là tiền đề thúc đẩy sự ham mê học hỏi và yêu thích nghành nghề của mình hơn và tiếp cận đến công nghệ một cách nhanh nhất.



Hình 1.3: Một số ứng dụng thực tế của Arduino

Bên cạnh đó, còn rất nhiều những ứng dụng khác của Arduino như :

* Điều khiển các thiết bị cảm biến âm thanh, ánh sáng.
* Làm máy in 3D.
* Làm đàn bằng ánh sáng.
* Làm lò nướng bánh biết tweet thông báo khi bánh đã chín.
* Arduino có khả năng đọc các thiết bị cảm biến, điều khiển động cơ,… Chính vì thế mà mã nguồn mở này được c dùng để làm bộ xử lý trung tâm của rất nhiều loại robot.
* Arduino còn có thể được sử dụng để tương tác với Joystick, màn hình,… khi chơi các game như Tetrix, phá gạch, Mario…
* Dùng để chế tạo ra máy bay không người lái.
* Điều khiển đèn giao thông, làm hiệu ứng đèn Led nhấp nháy trên các biển quảng cáo…
* Ngoài ra, Arduino còn rất nhiều ứng dụng hữu ích khác tùy thuộc vào sự sáng tạo của người sử dụng.
* Ví dụ: Muốn kết nối Internet thì có Ethernet shield, điều khiển động cơ thì có Motor shield, kết nối nhận tin nhắn thì có GSM shield,… Khá đơn giản, chỉ cần tập trung vào việc “lắp ghép” các thành phần này và sáng tạo ra các ứng dụng cần thiết là được.

# CHƯƠNG 2. THIẾT KẾ PHẦN CỨNG VÀ LẬP TRÌNH TRÊN ARDUINO IDE

## 2.1. Sơ đồ khối

Hệ thống bao gồm các thành phần chính sau :

* Khối cảm biến
* Khối hiện thị
* Khối điều khiển
* Khối nguồn

Trên hình 2.1 là sơ đồ kết nối của hệ thống:

Hình 2.1: Sơ đồ khối của hệ thống.

**Khối Hiển Thị LCD**

**Khối Điều Khiển**

**Khối Nguồn**

**Khối Cảm Biến**

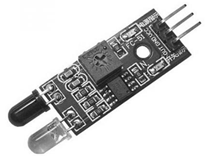
## 2.2. Chức năng của các khối

### 2.2.1. Khối cảm biến

Khối cảm biến sử dụng bộ cảm biến hồng ngoại – module FC-51 gồm một LED phát và một LED thu hồng ngoại hoạt động theo nguyên lý như sau:

LED phát hồng ngoại sẽ phát ra sóng ánh sáng có bước sóng hồng ngoại.

LED thu hồng ngoại bình thường có nội trở lớn, khi LED thu bị tia hồng ngoại chiếu vào thì nội trở giảm xuống tạo ra tín hiệu đưa đến khối điều khiển



Hình 2.2: Module FC-51 thu/phát hồng ngoại

*Trong đó : chân VCC nối với +5V trên Arduino*

*chân GND nối với 0V trên Arduino*

*chân OUT nối với chân 1-7 trên Arduino*

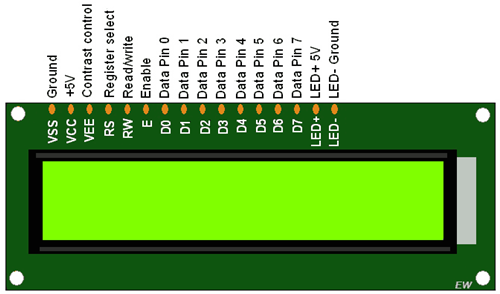
* Các thông số kỹ thuật của module thu/phát hồng ngoại FC-51:
* Sử dụng IC LM393.
* Góc mở: 350.
* Điện áp hoạt động: 3 – 6 V.
* Khoảng cách phát hiện: 2 – 30 cm (điều chỉnh bằng biến trở).
* Mức logic đầu ra: mức thấp khi có vật cản và mức cao khi không có vật cản.
* Dòng điện tiêu thụ: khoảng 23 mA (với mức điện áp 3,3 V) và khoảng 43 mA (với mức điện áp 5 V).
* Kích thước: dài 45 mm, rộng 14 mm và cao 7 mm.

### 2.2.2. Khối hiển thị

Hệ thống sử dụng màn hình LCD 16x2 và I2C để hiển thị giá trị đếm số lượng sản phẩm.

a. Đối với LCD 16×2 được sử dụng để hiển thị trạng thái hoặc các thông số :

* LCD 16×2 có 16 chân trong đó 8 chân dữ liệu (D0 – D7) và 3 chân điều khiển (RS, RW, EN).
* 5 chân còn lại dùng để cấp nguồn và đèn nền cho LCD 16×2.
* Các chân điều khiển giúp ta dễ dàng cấu hình LCD ở chế độ lệnh hoặc chế độ dữ liệu.
* Chúng còn giúp ta cấu hình ở chế độ đọc hoặc ghi.
* LCD 16×2 có thể sử dụng ở chế độ 4 bit hoặc 8 bit tùy theo ứng dụng ta đang làm.



Hình 2.3: Sơ đồ các chân kết nối của LCD 16x2

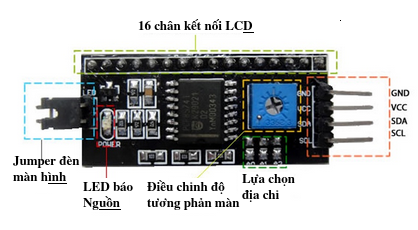
b. Đối với I2C

Vì LCD 16x2 có quá nhiều nhiều chân gây khó khăn trong quá trình đấu nối và chiếm dụng nhiều chân trên vi điều khiển. Module I2C ra đời và giải quyết vấn để này cho bạn.

Thay vì phải mất 6 chân vi điều khiển để kết nối với LCD 16×2 (RS, EN, D7, D6, D5 và D4) thì module IC2 bạn chỉ cần tốn 2 chân (SCL, SDA) để kết nối.

Module I2C hỗ trợ các loại LCD sử dụng driver HD44780(LCD 16×2, LCD 20×4, …) và tương thích với hầu hết các vi điều khiển hiện nay.

Trên hình 2.4 là sơ đồ chân của module I2C



Hình 2.4: Sơ đồ chân của I2C

*Trong đó: Chân 1 của I2C nối với GND (0V)*

*Chân 2 của I2C nối VCC (+5V)*

*Chân 3 (SDA) của I2C nối với A4 của Arduino*

*Chân 4 (SCL) của I2C nối với A5 của Arduino*

* THÔNG SỐ MẠCH CHUYỂN ĐỔI GIAO TIẾP I2C
* Kích thước: 41.5mm X19mm(W)X15.3MM(H)
* Trọng lượng: 5g
* Điện áp hoạt động: 2.5v-6v
* Jump chốt: Cung cấp đèn cho LCD hoặc ngắt
* Biến trở xoay độ tương phản cho LCD

### 2.2.3. Khối vi điều khiển và khối nguồn

a. Khối vi điều khiển

Khối vi điều khiển và khối nguồn được tích hợp trên module Arduino Uno R3. Khối vi điều khiển nhận tín hiệu từ khối cảm biến hồng ngoại (module FC-51), xử lý tín hiệu theo chương trình được nạp: tăng giá trị đếm và xuất hiển thị giá trị đếm lên màn hình LCD.

b. Khối nguồn

Khối nguồn cung cấp nguồn cho toàn bộ hệ thống thông thưởng được lấy thông qua nguồn DC 3.3-5V hoặc lấy nguồn trực tiếp từ Arduino.

## 2.3. Nguyên lý hoạt động

Nguyên lý hoạt động của mạch như sau:

- Tại thời điểm ban đầu chưa có vật cản đi qua, cảm biến hồng ngoại không nhận được tín hiệu do đó không có tín hiệu đưa đến vi điều khiển. Màn hình LCD trong trường hợp này hiển thị giá trị đếm nhớ trước đó.

- Khi có vật cản đi qua, cảm biến hồng ngoại thu nhận được tín hiệu và truyền tín hiệu đến vi điều khiển Arduino, vi điều khiển thực hiện tác vụ tăng giá trị đếm và xuất hiển thị giá trị đếm lên mà hình LCD đồng thời đảm bảo khi một vật đứng yên tại vị trí của cảm biến hồng ngoại thì số đếm không tăng lên và hiển thị lên LCD.

## 2.4. Phần mềm Arduino IDE

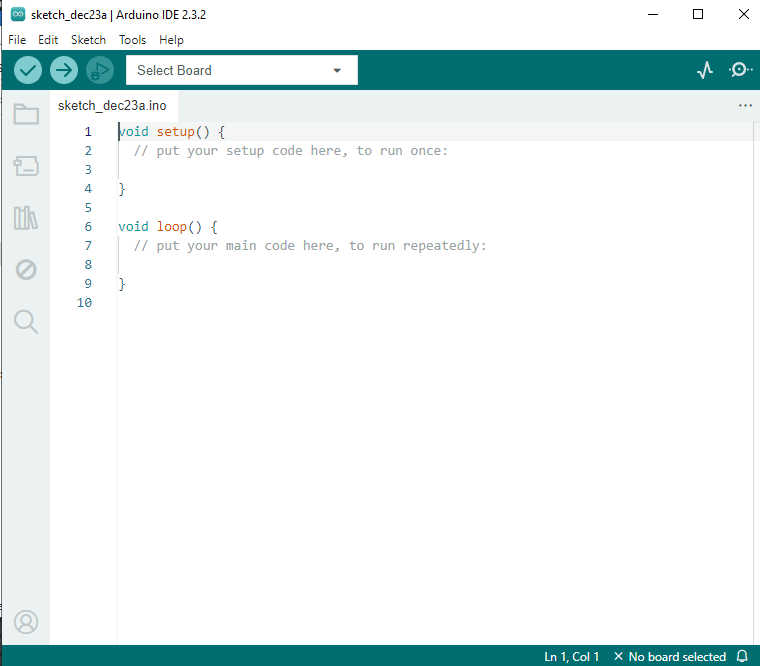
### 2.4.1. Giới thiệu phần mềm Arduino IDE

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. Nó bao gồm một trình soạn thảo mã với các tính năng như làm nổi bật cú pháp, khớp dấu ngoặc khối chương trình, thụt đầu dòng tự động và cũng có khả năng biên dịch và tải lên các chương trình vào board mạch với một nhấp chuột duy nhất. Một chương trình hoặc mã viết cho Arduino được gọi là "sketch".

Chương trình Arduino được viết bằng Choặc C++. Arduino IDE đi kèm với một thư viện phần mềm được gọi là "Wiring" từ dự án lắp ráp ban đầu, cho hoạt động đầu vào đầu ra phổ biến trở nên dễ dàng hơn nhiều. Người sử dụng chỉ cần định nghĩa hai hàm để thực hiện một chương trình điều hành theo chu kỳ.

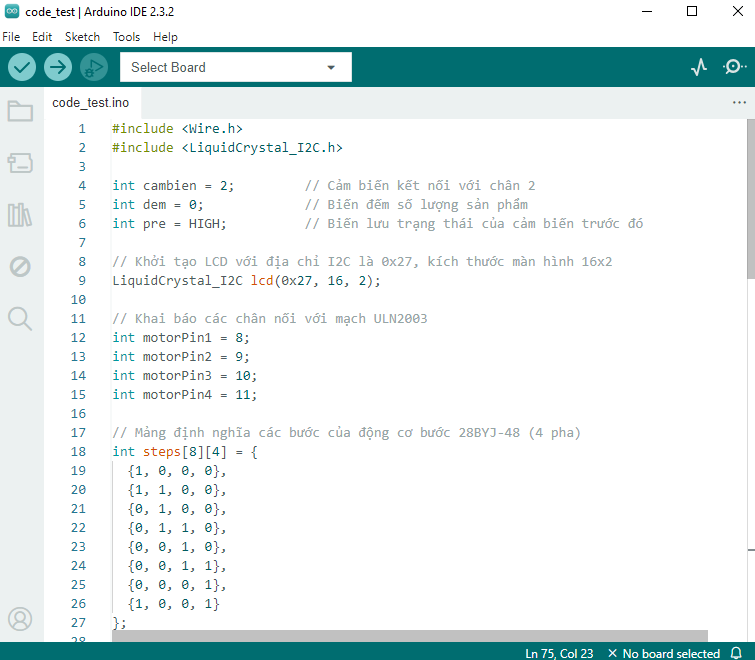
Khi các chúng ta bật điện bảng mạch Arduino, reset hay nạp chương trình mới, hàm setup() sẽ được gọi đến đầu tiên. Sau khi xử lý xong hàm setup(), Arduino sẽ nhảy đến hàm loop() và lặp vô hạn hàm này cho đến khi tắt điện board mạch Arduino.

Dưới đây là giao diện của phần mềm



Hình 2.5 Giao diện Phần mềm Arduino IDE

### 2.4.2. Lập trình trên Arduino IDE



**Kết quả thực nghiệm**

Trên hình 3.1 là kết quả thực tế của nhóm chúng em



Hình 3.1: Kết quả thực nghiệm

**CODE CHƯƠNG TRÌNH**

**Arduino IDE:**

#include <Wire.h>

#include <LiquidCrystal\_I2C.h>

int cambien = 2;         // Cảm biến kết nối với chân 2

int dem = 0;             // Biến đếm số lượng sản phẩm

int pre = HIGH;          // Biến lưu trạng thái của cảm biến trước đó

// Khởi tạo LCD với địa chỉ I2C là 0x27, kích thước màn hình 16x2

LiquidCrystal\_I2C lcd(0x27, 16, 2);

// Khai báo các chân nối với mạch ULN2003

int motorPin1 = 8;

int motorPin2 = 9;

int motorPin3 = 10;

int motorPin4 = 11;

// Mảng định nghĩa các bước của động cơ bước 28BYJ-48 (4 pha)

int steps[8][4] = {

  {1, 0, 0, 0},

  {1, 1, 0, 0},

  {0, 1, 0, 0},

  {0, 1, 1, 0},

  {0, 0, 1, 0},

  {0, 0, 1, 1},

  {0, 0, 0, 1},

  {1, 0, 0, 1}

};

void setup()

{

  Wire.begin();  // Khởi tạo I2C (không cần tham số vì sử dụng chân mặc định A4, A5)

  lcd.begin(16, 2);  // Khởi động LCD với kích thước 16x2

  lcd.backlight();  // Bật đèn nền LCD

  lcd.setCursor(0, 0);  // Đặt con trỏ tại vị trí dòng 0, cột 0

  lcd.print("DO AN NHOM 5");

  lcd.setCursor(1, 0);

  lcd.print("O An Nhom 5");

  delay(2000);  // Dừng 2 giây trước khi chuyển sang dòng tiếp theo

  lcd.clear();  // Xóa màn hình LCD

  lcd.setCursor(0, 1);  // Di chuyển con trỏ đến dòng 2, cột 0

  pinMode(cambien, INPUT);  // Đặt chân cảm biến là INPUT

  lcd.backlight();  // Bật đèn nền LCD

  // Khởi tạo các chân điều khiển động cơ bước là OUTPUT

  pinMode(motorPin1, OUTPUT);

  pinMode(motorPin2, OUTPUT);

  pinMode(motorPin3, OUTPUT);

  pinMode(motorPin4, OUTPUT);

  // Bắt đầu giao tiếp Serial để gửi dữ liệu đến máy tính

  Serial.begin(9600);

}

void loop()

{

  int giatricambien = digitalRead(cambien);  // Đọc giá trị từ cảm biến

  if (giatricambien == LOW && pre == HIGH)  // Khi cảm biến chuyển từ HIGH xuống LOW

  {

    dem = dem + 1;  // Tăng số lượng sản phẩm

    delay(200);  // Thêm một chút thời gian debounce

  }

  pre = giatricambien;  // Lưu trạng thái cảm biến hiện tại

  // Xóa dòng LCD cũ và hiển thị lại số lượng sản phẩm

  lcd.setCursor(0, 0);  // Di chuyển con trỏ về vị trí dòng 0, cột 0

  lcd.print("So luong: ");

  lcd.setCursor(10, 0);  // Di chuyển con trỏ để in số lượng

  lcd.print("  ");  // Xóa các ký tự cũ

  lcd.print(dem);  // In số lượng sản phẩm hiện tại

  // Gửi số lượng sản phẩm qua Serial Monitor (máy tính)

  Serial.print("So luong san pham: ");

  Serial.println(dem);

  // Lặp qua các bước để quay động cơ

  for (int i = 0; i < 10; i++) {  // Số bước để quay một vòng

    for (int j = 0; j < 8; j++) {

      // Cập nhật các chân điều khiển động cơ bước

      digitalWrite(motorPin1, steps[j][0]);

      digitalWrite(motorPin2, steps[j][1]);

      digitalWrite(motorPin3, steps[j][2]);

      digitalWrite(motorPin4, steps[j][3]);

      delay(1);  // Đợi một chút trước khi chuyển sang bước tiếp theo

    }

  }

}

**Python:**

import serial

# Kết nối với cổng COM của Arduino (thay COM3 bằng cổng của bạn)

ser = serial.Serial('COM3', 9600)

while True:

# Đọc dữ liệu từ Arduino

data = ser.readline().decode('utf-8').strip() # Đọc và giải mã dữ liệu

print(data) # Hiển thị dữ liệu nhận được từ Arduino

**KẾT LUẬN**

Trong suốt quá trình nghiên cứu và hoàn thiện sản phẩm, nhóm em thấy sản phẩm:

1. Mạch chạy ổn định
2. Sơ đồ đấu nối đơn giản
3. Độ nhạy của cảm biến cao
4. Khi có tín hiệu của cảm biến thì xuất lên màn hình LCD khá nhanh
5. Nhưng đôi khi gặp trục trặc về cảm biến do khoảng cách đếm hơi hạn chế chỉ khoảng 5cm mặc dù nhóm em đã căn chỉnh biến trở trên cảm biến.

### TÀI LIỆU THAM KHẢO

[1] “Arduino cho người mới bắt đầu” - tác giả IoT Maker Viet Nam,193 trang.

[2] “Vi Điều Khiển Và Ứng Dụng Arduino Dành Cho Người Tự Học” - Phạm Qunag Huy,Nguyễn Trọng Hiếu.