TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT THÀNH PHỐ HỒ CHÍ MINH

KHOA ĐIỆN ĐIỆN TỬ

BỘ MÔN KỸ THUẬT MÁY TÍNH - VIỄN THÔNG

-----------------------

ĐỒ ÁN MÔN HỌC 2

**TÊN ĐỀ TÀI**

**ỨNG DỤNG TỰ ĐỘNG HÓA TRONG QUY TRÌNH PHÂN LOẠI CÂN NẶNG SẢN PHẨM**

**NGÀNH CÔNG NGHỆ KỸ THUẬT MÁY TÍNH**

Sinh viên: **NGUYỄN THỊ NHẬT BĂNG** MSSV: 19119013

**LIÊU HOÀNG BÁ**

MSSV:15119007

Hướng dẫn**: PhD. Trương Ngọc Sơn**

TP. HỒ CHÍ MINH – 8/2023

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT THÀNH PHỐ HỒ CHÍ MINH

KHOA ĐIỆN ĐIỆN TỬ

BỘ MÔN KỸ THUẬT MÁY TÍNH - VIỄN THÔNG

-----------------------

ĐỒ ÁN MÔN HỌC 2

**TÊN ĐỀ TÀI**

**ỨNG DỤNG TỰ ĐỘNG HÓA TRONG QUY TRÌNH PHÂN LOẠI CÂN NẶNG SẢN PHẨM**

**NGÀNH CÔNG NGHỆ KỸ THUẬT MÁY TÍNH**

Sinh viên: **NGUYỄN THỊ NHẬT BĂNG**

MSSV: 19119013

**LIÊU HOÀNG BÁ**

MSSV:15119007

Hướng dẫn**: PhD. Trương Ngọc Sơn**

TP. HỒ CHÍ MINH – 8/2023

BẢNG NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

**LỜI CẢM ƠN**

“Lời đầu tiên, nhóm chúng em xin gửi lời cảm ơn chân thành đến Thầy Trương Ngọc Sơn, người đã đồng hành cùng nhóm chúng em trong suốt quá trình học phần Đồ Án 2 trong việc giúp đỡ, đánh giá và nhận xét những hạn chế trong đồ án này”.

Học phần Đồ Án 2 là môn học thú vị, bổ ích và có tính thực tế cao. Đảm bảo cung cấp đầy đủ kiến thức, kỹ năng, giúp sinh viên có thể ứng dụng vào thực tế. Tuy nhiên, do khả năng tiếp thu thực tế còn nhiều hạn hẹp, kiến thức lý thuyết chưa sâu rộng nhưng đó là kinh nghiệm quý báu trong hành trang để nhóm phát triển cho đồ án tốt nghiệp và xa hơn là trong công việc, sự nghiệp tương lai sau này. Mặc dù bản thân chúng em đã cố gắng hết sức hằng tuần nhưng chắc chắn đồ án này khó tránh khỏi những thiếu sót, kính mong thầy xem xét và góp ý để bài tiểu luận của nhóm được hoàn thiện và tốt hơn.Chúng em xin chân thành cảm ơn!

**MỤC LỤC**

**DANH MỤC HÌNH**

# Giới thiệu

## Giới thiệu

Ngày nay cùng với sự phát triển của nền kinh tế thị trường, khoa học kỹ thuật luôn luôn đóng góp một vai trò nhất định 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 sự ưu tiên hàng đầu. Lộ trình của sản phẩm từ khi được tạo ra đến khi tới tay khách hàng thường sẽ thông qua đơn vị vận chuyển, vận chuyển là nhu cầu thiết yếu của sự phát triển kinh tế nhằm mục đích thay đổi vị trí của hàng hóa và con người từ nơi này đến nơi khác bằng các phương tiện vận chuyển. Trong kinh doanh thương mại vận chuyển hàng hóa có vai trò đặc biệt quan trọng. Chúng ta có thể thấy là ngày nay các sàn thương mại điện tử như Shopee, Lazada, TiKi,… họ bán rất nhiều sản phẩm, mỗi sản phẩm sẽ có khối lượng khác nhau dẫn đến tiền vận chuyển hàng khác nhau nên cần phải phân loại sản phẩm theo từng khoảng cân nặng.

Với sự ra đời và phổ biến rộng rãi của băng tải công nghiệp, chúng ta gần như có thể tối ưu hóa mọi lĩnh vực. Với mục đích vận dụng những công nghệ khoa học kỹ thuật tiên tiến vào trong quá trình sản xuất nhằm giảm thiểu sức lao động của con người và nâng cao năng suất để có thể tạo ra một hệ thống phân loại sản phẩm theo dây chuyền, thân thiện với người tiêu dùng, khả năng tùy biến cao và giá thành phải chăng, nhóm em chọn đề tài: " Ứng dụng tự động hóa trong quy trình phân loại cân nặng sản phẩm"

## mục tiêu đề tài

Mục tiêu đặt ra là nghiên cứu thiết kế hệ thống phân loại sản phẩm theo cân nặng có kiểu dáng nhỏ gọn, dễ dàng lắp đặt, bảo trì, sửa chữa.

Các vấn đề cần được giải quyết đó là:

- Nắm rõ nguyên lý hoạt động của cảm biến cân nặng

- Phân tích tính toán và lựa chọn vật liệu, thông số kỹ thuật của các chi tiết sao cho thỏa mãn yêu cầu của đề tài: nhỏ, gọn, nhẹ, bền, có tính thẩm mỹ cao, dễ dàng lắp đặt và sửa chữa.

- Điều khiển tự động.

- Đảm bảo an toàn cho người sử dụng và sản phẩm không bị hỏng.

Với các chức năng:

- Chức năng 1: Điều chỉnh tốc độ băng tải

- Chức năng 2: Đo được cân nặng bằng Loadcell kết nối module HX711 để chuyển tín hiệu điện áp sang tín hiệu số

- Chức năng 3: Phân loại sản phẩm chính xác theo tiêu chuẩn từng khoảng cân nặng

- Chức năng 4: Đếm số lượng sản phẩm

- Chức năng 5: Hiển thị và cập nhật lên lcd, firebase, web

## GIỚI HẠN ĐỀ TÀI

Hệ thống phân loại sản phẩm là một đề tài đã được nghiên cứu và phát triển từ lâu. Hiện nay trong các nhà máy xí nghiệp có rất nhiều hệ thống hoàn thiện cả về chất lượng và thẩm mỹ. Tuy nhiên, trong phạm vi một đề tài nghiên cứu, với những giới hạn về kiến thức, thời gian và kinh phí đề tài giới hạn:

- Cần đặt cân ở vị trí cân bằng để đảm bảo việc đo đạt chính xác

- Nghiên cứu, điều khiển hệ thống phân loại sản phẩm dùng Esp32, động cơ Servo để gạt sản phẩm, cảm biến hồng ngoại đếm sản phẩm và một số thiết bị cần thiết khác.

- Về chức năng: Hệ thống chỉ phân loại sản phẩm theo cân nặng

## PHƯƠNG PHÁP NGHIÊN CỨU

Các phương pháp được sử dụng để giải quyết đề tài: Phương pháp tổng hợp tài liệu lý thuyết, phương pháp khảo sát hệ thống thực tế, phân tích thiết kế và lập trình cho hệ thống, thử nghiệm và vận hành cho hệ thống

## ĐỐI TƯỢNG VÀ PHẠM VI NGHIÊN CỨU

***Đối tượng nghiên cứu:***

Tìm hiểu và mô phỏng trên ESP32 cùng với Loadcell hx711, cảm biến hồng ngoại, LCD, Servo, Stepper motor …

***Phạm vi nghiên cứu:***

- Các thông số kỹ thuật, sơ đồ chân, nguyên lí hoạt động, các chuẩn giao tiếp của các thiết bị

- Đo và hiển thị cân nặng lên lcd, firebase, web

- Sử dụng servo để đẩy các sản phẩm theo từng khoảng cân nặng

- Đếm sản phẩm bằng cảm biến hồng ngoại

## BỐ CỤC QUYỂN BÁO CÁO

Quyền báo cáo gồm 5 chương:

Chương 1: Giới thiệu

Chương 2: Cơ sở lý thuyết

Chương 3: Thiết kế hệ thống

Chương 4: Kết quả

Chương 5: Kết luận và hướng phát triển

# 

# CƠ SỞ LÝ THUYẾT

## ESP32

ESP32 là một hệ thống vi điều khiển trên chip (SoC) giá rẻ của Espressif Systems, nhà phát triển của ESP8266 SoC. Nó là sự kế thừa của SoC ESP8266 và có cả hai biến thể lõi đơn và lõi kép của bộ vi xử lý 32-bit Xtensa LX6 của Tensilica với Wi-Fi và Bluetooth tích hợp. Giống như ESP8266, các thành phần RF tích hợp của ESP32 như bộ khuếch đại công suất, bộ khuếch đại nhận tiếng ồn thấp, công tắc ăng-ten, bộ lọc và Balun RF.

1. **Thông số kỹ thuật**

- Bộ vi xử lý LX6 32-bit lõi đơn hoặc lõi kép với xung nhịp lên đến 240 MHz.

- 520 KB SRAM, 448 KB ROM và 16 KB SRAM RTC.

- Hỗ trợ kết nối Wi-Fi 802.11 b / g / n với tốc độ lên đến 150 Mbps.

- Hỗ trợ cho cả thông số kỹ thuật Bluetooth v4.2 và BLE cổ điển.

- 18 kênh SAR ADC 12 bit và 2 kênh DAC 8 bit

- Ethernet MAC cho giao tiếp mạng LAN vật lý (yêu cầu PHY bên ngoài).

- 1 bộ điều khiển host cho SD / SDIO / MMC và 1 bộ điều khiển slave cho SDIO/ SPI.

- Động cơ PWM và 16 kênh LED PWM.

- Khởi động an toàn và mã hóa Flash.

1. **Sơ đồ**

Gồm: 34 GPIO có thể lập trình, 18 kênh ADC 12 bit, 2 kênh DAC 8-bit, 16 kênh PWM, 3 giao diện UART, 3 giao diện SPI, 2 Giao diện I2C, 2 Giao diện I2S, 10 GPIO cảm ứng điện dung, 16 GPIO RTC



*Hình 2.1 Sơ đồ chân ESP32*

1. **Chuẩn giao tiếp**

**Chuẩn I2C:** Gồm 2 dòng dữ liệu SDA và SCL. Giao tiếp I2C gồm 1 master và nhiều slave. Mỗi Slave sẽ có một địa chỉ để giao tiếp với master.

SDA: đường dữ liệu.

SCL: đường xung clock

ESP32 có hai kênh I2C và bất kỳ chân nào cũng có thể được đặt làm SDA hoặc SCL. Khi sử dụng ESP32 với Arduino IDE, các chân I2C mặc định là: GPIO 21 (SDA), GPIO 22 (SCL)

## LCd

LCD 1604 được sử dụng trong rất nhiều các ứng dụng của vi điều khiển, có rất nhiều ưu điểm so với các dạng hiển thị khác như: khả năng hiển thị kí tự đa dạng (chữ, số, kí tự đồ họa); dễ dàng đưa vào mạch ứng dụng theo nhiều giao thức giao tiếp khác nhau

1. **Thông số kỹ thuật**

Kích thước: 86,96 x 60 x 13mm

Diện tích hiển thị: 62,3 x 27,6mm

Số lượng ký tự: 16 x 4

Điện áp ổ đĩa: nguồn 3,3V / 5V

Nhiệt độ làm việc: -20 độ đến +70 độ

Nhiệt độ lưu trữ: -30 độ đến +80 độ

1. **Sơ đồ**

Chân D0→D7 chân data.

VSS: chân GND.

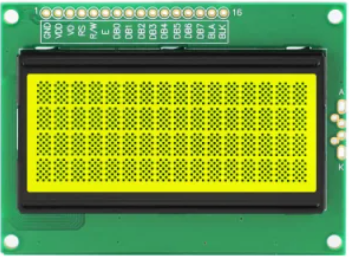
VDD: chân dương 5V

VEE: điều chỉnh độ sáng màn hình LCD bằng cách điều chỉnh điện trở.

R/W: Read/Write.

E: cho phép mô-đun.

RS: Chọn thanh ghi.



*Hình 2.2 Sơ đồ chân LCD*

1. **Chuẩn giao tiếp**

**Chuẩn I2C:** ESP32 làm Master, LCD làm Slave giao tiếp với nhau qua

địa chỉ 0x27, 0x3F.

## **CHUYỂN ĐỔI I2C GIAO TIẾP LCD1604**

LCD có quá nhiều chân gây khó khăn trong quá trình kết nối và chiếm dụng nhiều chân của vi điều khiển nên khi sử dụng module chuyển đổi I2C cho LCD sẽ giải quyết vấn đề này, thay vì sử dụng tối thiểu 6 chân của vi điều khiển để kết nối với LCD (RS, EN, D7, D6, D5 và D4) thì với module chuyển đổi bạn chỉ cần sử dụng 2 chân (SCL, SDA) để kết nối.

1. **Thông số kỹ thuật**

Điện áp hoạt động: 2.5-6V DC

Hỗ trợ màn hình: LCD1602,1604,2004 (driver HD44780)

Kích thước: 41.5mm(L)x19mm(W)x15.3mm(H)

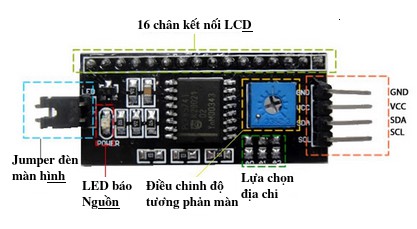
Trọng lượng: 5g

Tích hợp Jump chốt để cung cấp đèn cho LCD hoặc ngắt

Tích hợp biến trở xoay điều chỉnh độ tương phản cho LCD

1. **Sơ đồ**

Gồm: 16 chân kết nối với LCD, SDA và SCL chân giao tiếp I2C



*Hình 2.3 Sơ đồ chân module chuyển đổi I2C giao tiếp LCD*

1. **Chuẩn giao tiếp:** I2C

## **LOADCELL**

Loadcell hay còn được gọi là cảm biến trọng tải, thiết bị này được dùng để chuyển đổi lực mô-men xoắn hoặc trọng lượng thành tín hiệu điện. Cảm biến loadcell thường được ứng dụng trong các loại cân điện tử như: cân bàn, cân điện tử công nghiệp, cân treo, cân kỹ thuật, cân băng tải…

1. **Thông số kỹ thuật**

Điện áp: giá trị điện áp hoạt động của loadcell (5 đến 15 V).

Nhiệt độ hoạt động -20 ~ 65 độ C

Trở kháng đầu vào: 1066 +- 20 (Ω )

Trở kháng ngõ ra 1000 +- 20 (Ω )

Trở kháng cách li 2000 (MΩ)

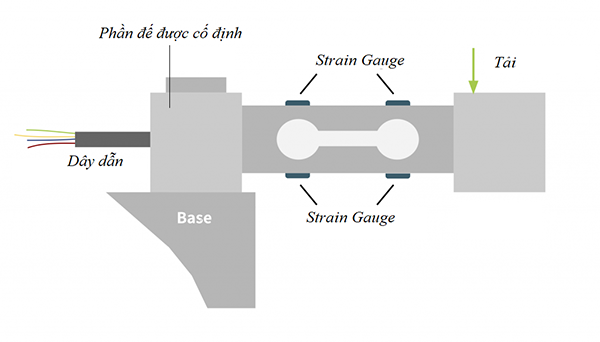
1. **Sơ đồ**

Dây đỏ: ngõ vào ( + )

Dây đen: ngõ vào ( – )

Dây xanh lá, ngõ ra ( + )

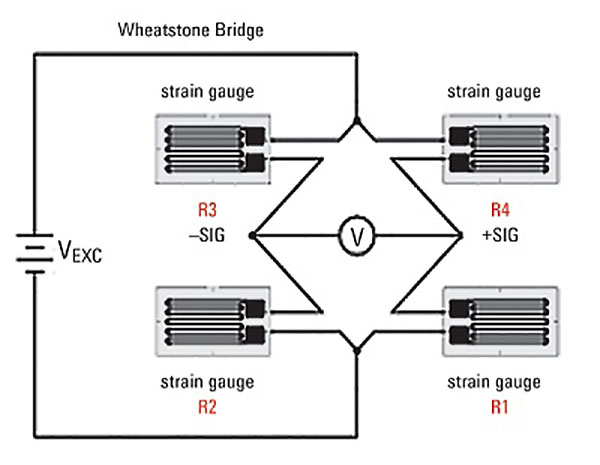
Dây trắng: ngõ ra ( – )



*Hình 2.4.1 Sơ đồ chân Loadcell*

1. **Nguyên lý hoạt động**

Loadcell hoạt động dựa theo nguyên lý Wheatstone (nguyên lý cầu điện trở cân bằng).



*Hình 2.4.2 Nguyên lý Loadcell*

Khi có một lực tác động lên loadcell, thân loadcell sẽ bị giãn ra hoặc nén vào. Từ đó, chiều dài sợi kim loại strain gauges dán trên thân loadcell thay đổi khiến giá trị của các điện trở thay đổi theo. Điện áp đầu ra tiếp tục thay đổi, dữ liệu này sẽ được chuyển thành dạng số nhờ bộ khuếch đại của cân điện tử.

## **HX711**

Mạch chuyển đổi ADC 24bit Loadcell HX711: module chuyển đổi analog sang digital 24-bit, được sử dụng để đọc giá trị điện trở thay đổi từ cảm biến Loadcell (thường rất nhỏ không thể đọc trực tiếp bằng VĐK) với độ phân giải ADC 24bit và chuyển sang giao tiếp 2 dây (Clock và Data) để gửi dữ liệu về Vi điều khiển, thích hợp để sử dụng với Loadcell trong các ứng dụng đo cân nặng. Mạch chuyển đổi ADC HX711 không chỉ có một vài chức năng cơ bản, cũng có tích hợp cao, phản ứng nhanh, khả năng chống nhiễu, và độ tin cậy cao.

1. **Thông số kỹ thuật**

- Điện áp hoạt động : 2.7 - 5V

- Dòng tiêu thụ : < 1.5 mA

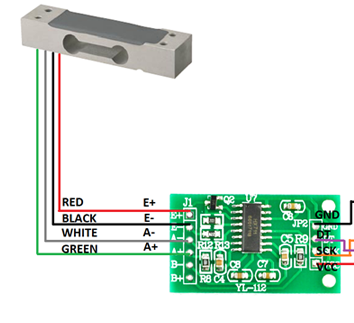
- Tốc độ lấy mẫu : 10 - 80 SPS ( tùy chỉnh )

- Độ phân giải : 24 bit ADC

- Độ phân giải điện áp : 40mV

- Kích thước : 38 x 21 x 10 mm

1. **Sơ đồ**



*Hình 2.5 Sơ đồ chân HX711*

1. **Nguyên lý hoạt động:**

Khi đặt cân lên cảm biến lực, nó tạo ra điện áp analog ở các chân đầu ra phụ thuộc vào trọng lượng của tải. Kênh A đọc đầu ra cảm biến thông qua bộ khuếch đại và bộ ghép kênh đầu vào. Bộ khuếch đại điều chỉnh độ lớn tín hiệu đầu vào và chuyển tín hiệu này đến mạch chuyển đổi tín hiệu digital. Mạch giao tiếp digital chuyển tín hiệu đầu vào analog sang tín hiệu digital 24 bit và gửi tín hiệu ra chân đầu ra DOUT.

## **SERVO SR90**

Động Cơ Servo SG90 là loại động cơ được dùng phổ biến trong các mô hình điều khiển nhỏ và đơn giản như cánh tay robot. Động cơ có tốc độ phản ứng nhanh, được tích hợp sẵn Driver điều khiển động cơ, dễ dàng điều khiển góc quay bằng phương pháp điều độ rộng xung PWM. Động Cơ Servo SG90 được sử dụng rất nhiều trong thiết kế các loại mô hình.

1. **Thông số kỹ thuật**

Điện áp hoạt động: 4.8-5VDC

Tốc độ: 0.12 sec/ 60 deg (4.8VDC)

Lực kéo: 1.6 Kg.cm

Kích thước: 21x12x22mm

Trọng lượng: 9g.

1. **Sơ đồ**

- Màu nâu: dây nối đất nối với đất của hệ thống

- Màu đỏ: nguồn điện động cơ thường + 5V được sử dụng

- Màu cam: tín hiệu PWM được đưa vào qua dây này để điều khiển động cơ



*Hình 2.6 Sơ đồ chân Servo SG90*

1. **Nguyên lí hoạt động**

Tín hiệu PWM tạo ra nên có tần số 50Hz và chu kỳ PWM phải là 20ms. Trong đó Thời gian đúng giờ có thể thay đổi từ 1ms đến 2ms. Vì vậy, khi on-time là 1ms, động cơ sẽ ở 0 ° và khi 1.5ms, động cơ sẽ ở 90 °, tương tự khi nó là 2ms, nó sẽ là 180 °. Vì vậy, bằng cách thay đổi thời gian đúng giờ từ 1ms đến 2ms, động cơ có thể được điều khiển từ 0 ° đến 180 °

## **STEP MOTOR**

Step Motor thực chất là một động cơ đồng bộ dùng để biến đổi các tín hiệu điều khiển dưới dạng các xung điện rời rạc kế tiếp nhau thành các chuyển động góc quay hoặc các chuyển động của Rotor và có khả năng cố định Rotor vào những vị trí cần thiết.

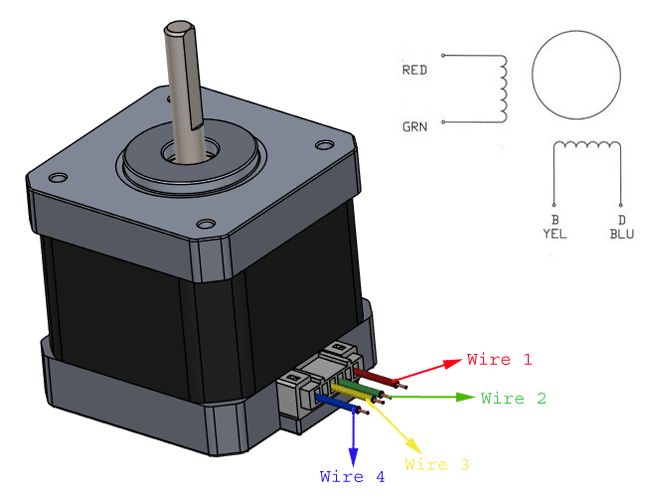
1. **Thông số kỹ thuật**

- Độ phân giải: 1.8 độ, 200 xung/ 1 vòng.

- Kích thước: 42x42x35 (DxRxC).

- Dòng: 1.2 -1,4A tùy lô.

1. **Sơ đồ**



1. **Nguyên lý hoạt động**

Khi bạn cấp điện vào trong cuộn dây của stato, cuộn dây này sẽ đóng vai trò chính là một nam châm điện. Nó sẽ gây ra lực tác động, từ đó làm quay roto. Step Motor làm việc được là nhờ có bộ chuyển mạch điện tử đưa các tín hiệu điều khiển vào stato theo một thứ tự và một tần số nhất định. Tổng số góc quay của roto tương ứng với số lần chuyển mạch, cũng như chiều quay và tốc độ quay của roto, phụ thuộc vào thứ tự chuyển đổi và tần số chuyển đổi. Khi một xung điện áp đặt vào cuộn dây stato (phần ứng) của động cơ bước thì roto (phần cảm) của động cơ sẽ quay đi một góc nhất định, góc ấy là một bước quay của động cơ. Khi các xung điện áp đặt vào các cuộn dây phần ứng thay đổi liên

tục thì roto sẽ quay liên tục (Nhưng thực chất chuyển động đó vẫn là theo các bước rời rạc).

# thiết kế hệ thống

## THIẾT KẾ PHẦN CỨNG

### Yêu cầu của hệ thống

- Hệ thống bao gồm có những chức năng sau:

Đo được giá trị cân nặng của sản phẩm

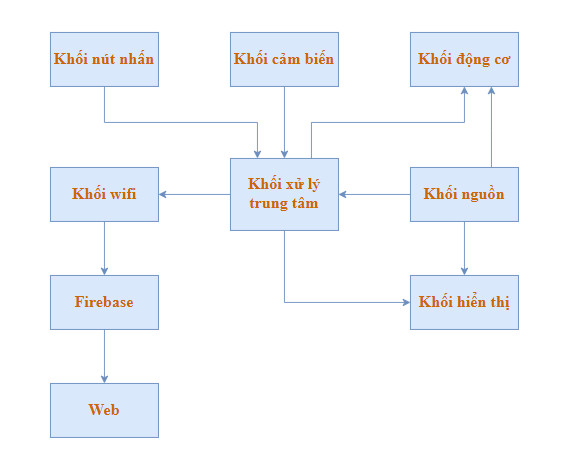
Điều khiển động cơ trên băng tải hoạt động ở chế độ tự động

Hiển thị cân nặng của sản phẩm và chế độ hoạt động của băng tải trên màn hình LCD

Dùng động cơ để phân loại sản phẩm dựa trên cân nặng

Kết nối wifi và cập nhật các dữ liệu liên quan lên Firebase, Web.

### Sơ đồ khối phần cứng



**Khối xử lý trung tâm:** Xử lý các dữ liệu nhận được từ khối cảm biến và truyền dữ liệu đến khối hiển thị. Nhận tín hiệu điều khiển từ khối nút nhấn sau đó gửi tín hiệu điều khiển cho khối động cơ. Kết nối mạng wifi để cập nhật các dữ liệu thông qua khối wifi.

**Khối cảm biến:** Thu thập dữ liệu từ cảm biến và gửi dữ liệu đến khối xử lý trung tâm.

**Khối nút nhấn:** Gửi tín hiệu điều khiển cho khối xử lý trung tâm.

**Khối động cơ:** Nhận tín hiệu điều khiển từ khối xử lý trung tâm để xử lý các chức năng điều khiển động cơ.

**Khối hiển thị:** Hiển thị dữ liệu nhận được từ khối xử lý trung tâm.

**Khối wifi:** Kết nối với mạng wifi và nhận dữ liệu từ khối xử lý trung tâm.

**Firebase:** Là cơ sở dữ liệu của hệ thống nhằm lưu trữ và cập nhật các thông tin dữ liệu liên quan.

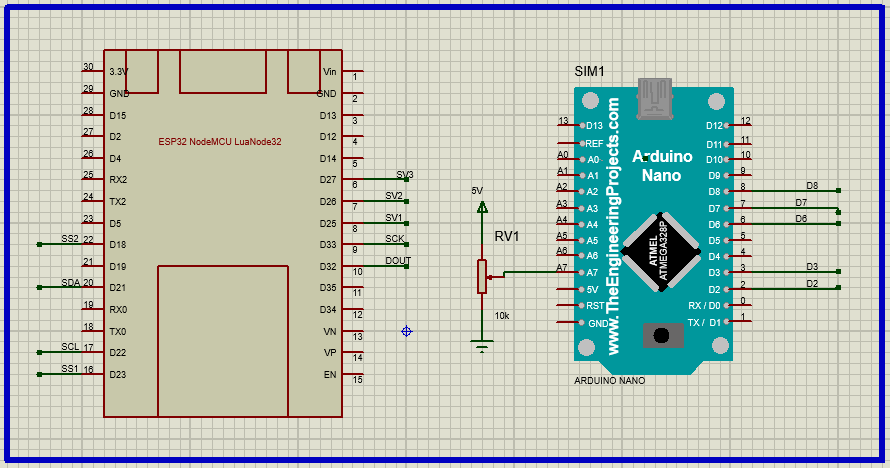
**Web:** Hiển thị dữ liệu sản phẩm

**Khối nguồn:** Cung cấp nguồn hoạt động ổn định cho các khối: khối xử lý trung tâm, khối động cơ, khối hiển thị và khối cảm biến

### Thiết kế từng khối

**Khối xử lý trung tâm:**

Nhóm em sử dụng vi điều khiển ESP32.



*Hình 3.1 Sơ đồ nguyên lí khối xử lí trung tâm*

- Sơ đồ kết nối chân của ESP32 với module chuyển đổi I2C

Chân D22 của ESP32 nối với chân SCL của mạch chuyển đổi I2C

Chân D21 của ESP32 nối với chân SDA của mạch chuyển đổi I2C

- Sơ đồ kết nối chân của ESP32 với module Hx711

Chân D32 của ESP32 nối với chân DOUT của Hx711

Chân D33 của ESP32 nối với chân SCK của Hx711

- Sơ đồ kết nối chân của ESP32 với Servo

Chân D25, D26, 27 của ESP32 nối với dây cam của Servo SG90

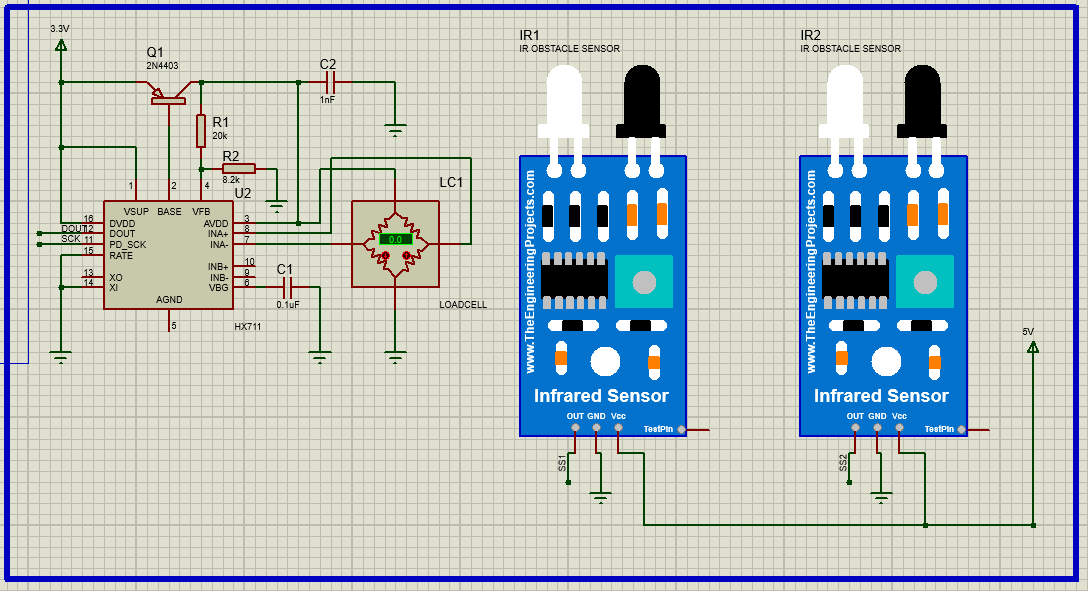
- Sơ đồ kết nối chân của ESP32 với nút nhấn đơn

Vi điều khiển ESP32 đóng vai trò là bộ xử lý trung tâm của hệ thống. Gửi tín hiệu điều khiển đến các module động cơ để thực hiện chức năng. Nhận tín hiệu dữ liệu từ cảm biến, xử lý dữ liệu và gửi đến module LCD để hiển thị. Cập nhật các dữ liệu đó lên Firebase thông qua giao tiếp Wifi. Ngoài ra, vi điều khiển còn cung cấp nguồn hoạt động 3,3-5V đến các module Hx711, mạch chuyển đổi giao tiếp I2C.

**Khối cảm biến:**

Dựa theo yêu cầu của hệ thống, nhóm em quyết định chọn và sử dụng cảm biến hồng ngoại để đếm sản phẩm và cảm biến Load cell weight sensor 1Kg để đo cân nặng với độ chính xác cao, sai số thấp. Bên cạnh đó, dùng mạch chuyển đổi tín hiệu ADC 24bit Hx711 để khuếch đại điện áp từ cảm biến cân nặng load cell.

Cảm biến cân nặng load cell là cảm biến có thể chuyển đổi trọng lượng thành một tín hiệu điện. Giá trị tác dụng tỉ lệ với sự thay đổi điện trở cảm ứng trong cầu điện trở, từ đó trả về tín hiệu điện áp tương ứng.



*Hình 3.2 Sơ đồ nguyên lí khối cảm biến*

- Sơ đồ kết nối chân của Hx711 và Loadcell:

Dây màu đỏ của Load cell nối với E+ của Hx711

Dây màu đen của Load cell nối với E- của Hx711

Dây màu xanh lá của Load cell nối với A+ của Hx711

Dây màu trắng của Load cell nối với A- của Hx711

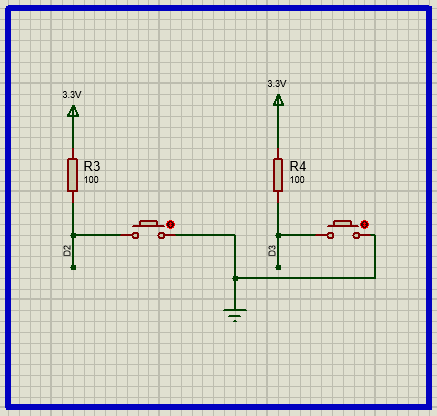
Chân VCC của Hx711 nối với điện áp 3.3V của ESP32

Chân GND của Hx711 nối với GND của ESP32

Chân DOUT của Hx711 nối với chân D32 của ESP32

Chân SCK nối Hx711 với chân D33 của ESP32

**Khối nút nhấn:**

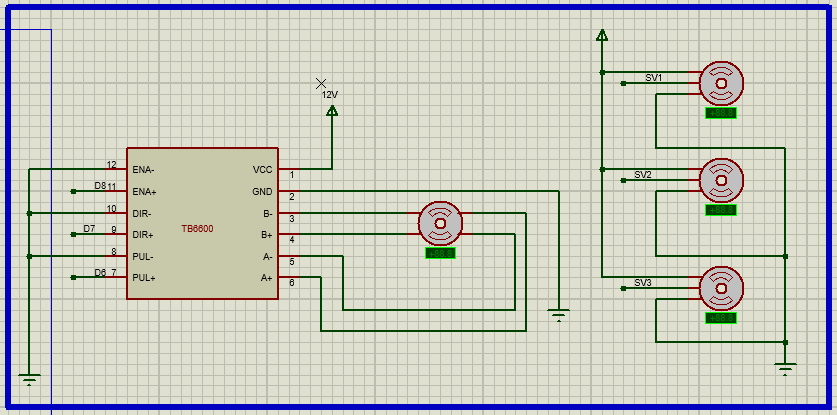


*Hình 3.3 Sơ đồ nguyên lí khối nút nhấn*

Sử dụng nút nhấn đơn 4 chân với chức năng chạy và dừng băng tải

Chân D2, D3 nối với chân D2, D3 của Arduino Nano

**Khối động cơ:**



*Hình 3.4 Sơ đồ nguyên lí khối động cơ*

Nhóm sử dụng động cơ Servo SG90 để làm cần gạt phân loại sản phẩm cho hệ thống, dùng động cơ bước kết nối với module điều khiển động cơ TB6600 để thực hiện chức năng của băng tải.

- Sơ đồ kết nối của motor Servo SG90

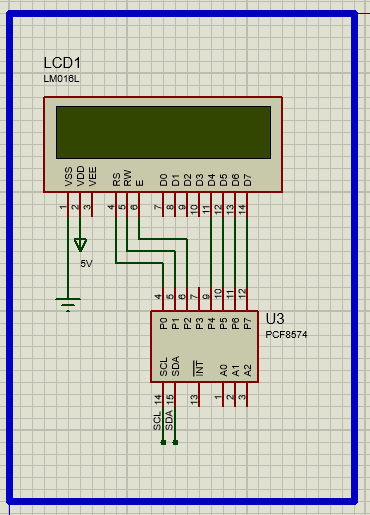
Dây màu đỏ nối với chân 3,3V

Dây màu đen nối với chân GND

Dây màu cam nối với chân D25, D26, D27

**Khối hiển thị:**

Nội dung cần hiển thị là khối lượng, số lượng của sản phẩm và chế độ hoạt động của băng tải. Nhóm em chọn màn hình LCD 16x4 để phù hợp với chức năng hiển thị ký tự số và chữ viết. Do kết nối giữa vi điều khiển với màn hình LCD cần sử dụng nhiều chân điều khiển, nhóm quyết định dùng mạch chuyển đổi giao tiếp I2C cho LCD để tiết kiệm số chân điều khiển.



*Hình 3.5 Sơ đồ nguyên lí khối hiển thị*

Sơ đồ kết nối chân của mạch chuyển đổi giao tiếp I2C:

Chân VCC kết nối với chân 5V của Arduino

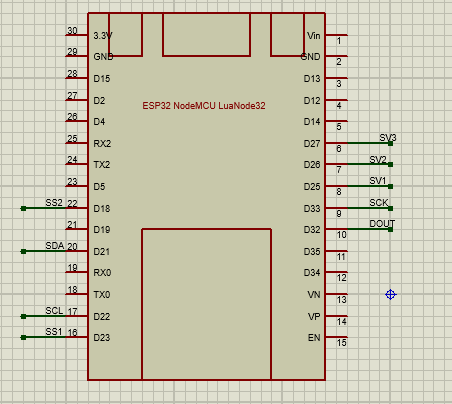
Chân GND kết nối với chân GND của Arduino

Chân SCL nối với chân D22 của Arduino

Chân SDA nối với chân D21 của Arduino

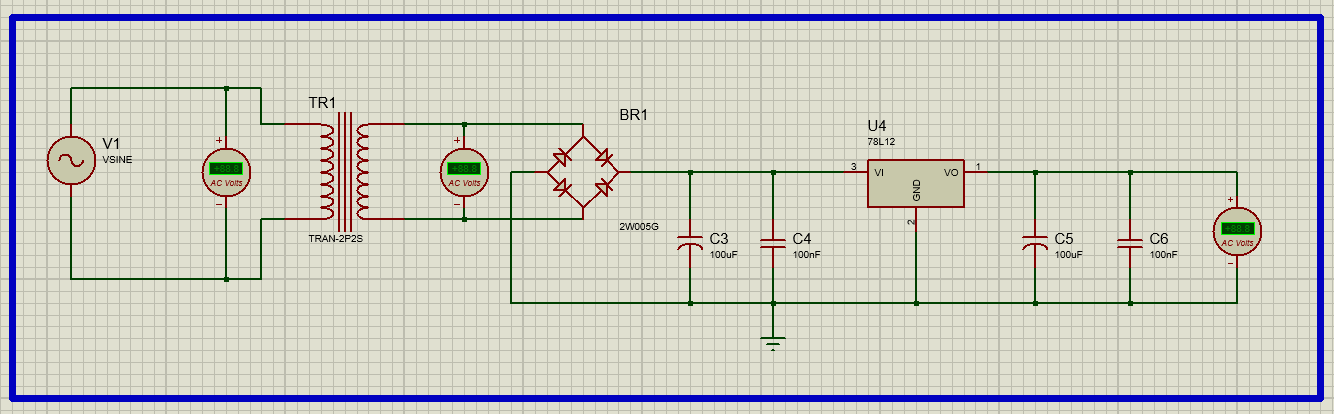
**Khối wifi:**

Chọn module ESP32 để thực hiện chức năng kết nối wifi



*Hình 3.6 Sơ đồ nguyên lí khối wifi*

**Khối Nguồn:**



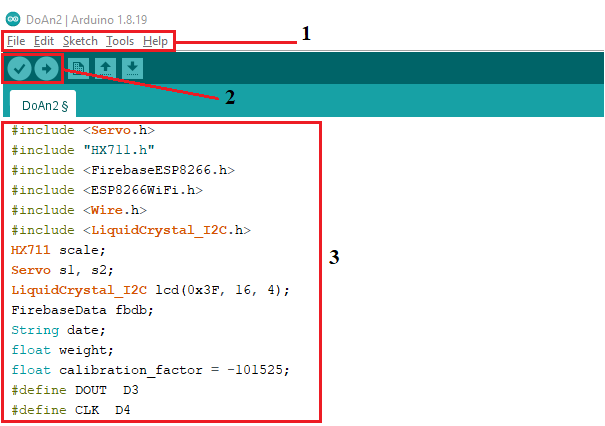
*Hình 3.7 Sơ đồ nguyên lí khối nguồn*

Nhóm sử dụng nguồn Adapter 12V

## THIẾT KẾ PHẦN MỀM

### Chức năng hoạt động của phần mềm

Nhóm sử dụng phần mềm Arduino IDE để lập trình cho vi điều khiển.



*Hình 3.8 Hướng dẫn sử dụng cơ bản phần mềm IDE*

Giao diện của phần mềm Arduino IDE có nhiều phần. Tuy nhiên, chỉ cần chú trọng vào những phần quan trọng như hình trên

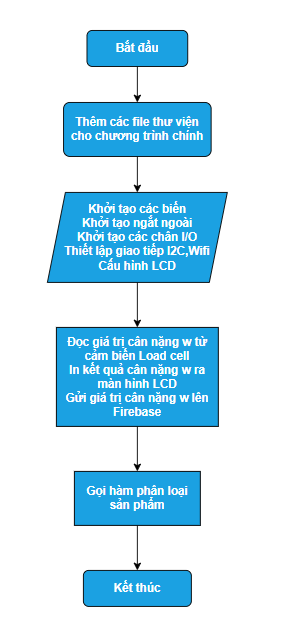
- Phần 1: Menu lệnh: Dùng để thêm thư viện, lưu, tạo project mới, chọn vi điều khiển cần lập trình...

- Phần 2: Nút kiểm tra và nạp chương trình: Dùng để biên dịch chương trình và kiểm tra lỗi. Sau khi đã hoàn tất quá trình trên, chương trình sẽ được nạp vào vi điều khiển đã được thiết lập trong Tool ở phần 1.

- Phần 3: Đoạn mã của chương trình chính.

### Lưu đồ hoạt động của hệ thống

Lưu đồ của chương trình chính



*Hình 3.9 Lưu đồ chương trình chính*

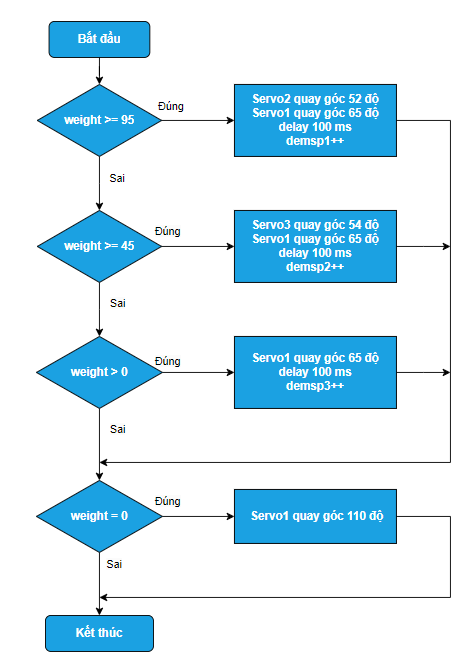
Giải thích lưu đồ:

Chương trình chính bắt đầu với những bước khởi tạo các biến số, khởi tạo hàm ngắt, thiết lập các chân ngõ vào ra để giao tiếp với các module khác. Gọi hàm cấu hình cho module LCD, hàm liên kết với Firebase, kết nối wifi với vi điều khiển.

Tiếp theo, đọc giá trị cân nặng của sản phẩm từ cảm biến load cell và gán giá trị này với biến weight . Sau đó, hiển thị giá trị của weight lên màn hình LCD đồng thời gửi giá trị đó lên Firebase để cập nhật trên Web.

Để phân loại sản phẩm dựa trên giá trị cân nặng weight thu được từ cảm biến, sử dụng chương trình con phân loại sản phẩm.

Lưu đồ của chương trình con phân loại sản phẩm



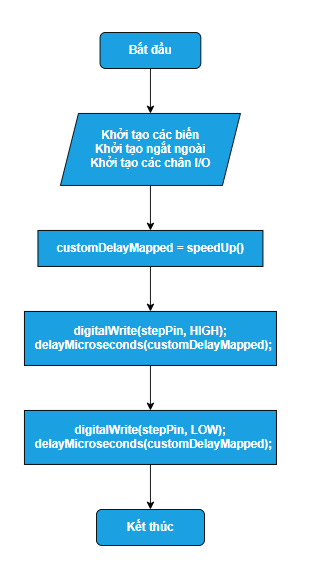
*Hình 3.10 Lưu đồ chương trình con phân loại, đếm sản phẩm*

Giải thích lưu đồ:

Chương trình phân loại sản phẩm sẽ điều khiển các động cơ Servo SG90 xoay đến các tọa độ đã được lập trình để làm cần gạt đưa sản phẩm đến đúng vị trí cần phân loại. Có 3 loại sản phẩm được hệ thống phân loại theo khối lượng, sản phẩm loại 1 có khối lượng trên 95 gram, sản phẩm loại 2 có khối lượng trên 45 gram và cuối cùng là sản phẩm loại 3 với khối lượng dưới 45 gram.

Khi đặt sản phẩm lên dĩa cân của cảm biến Load cell, chương trình sẽ điều khiển động cơ servo1 đẩy sản phẩm lên băng tải, đồng thời điều khiển động cơ servo2 làm cần gạt đưa sản phẩm đến vị trí mong muốn phân loại nếu giá trị của biến weight lớn hoặc bằng 95. Tiếp theo là tăng biến demsp1 lên một đơn vị để đếm số lượng sản phẩm loại 1. Khi giá trị của biến weight bé hơn 95 sẽ tiếp tục so sánh với 45 và áp dụng quy trình tương tự với động cơ servo3. Trường hợp còn lại với giá trị weight từ lớn hơn 0 đến 45 sẽ ứng với sản phẩm loại 3. Sau khi động cơ servo1 đã đẩy sản phẩm lên băng tải, khối lượng đọc được từ cảm biến lúc này sẽ bằng 0 và servo1 được điều khiển xoay về vị trí cũ.

Lưu đồ của chương trình điều khiển băng tải



*Hình 3.11 Lưu đồ chương trình điều khiển băng tải*

Giải thích lưu đồ:

Khởi tạo hàm ngắt và các chân GPIO ngõ ra để điều khiển tín hiệu cho động cơ bước của băng tải.

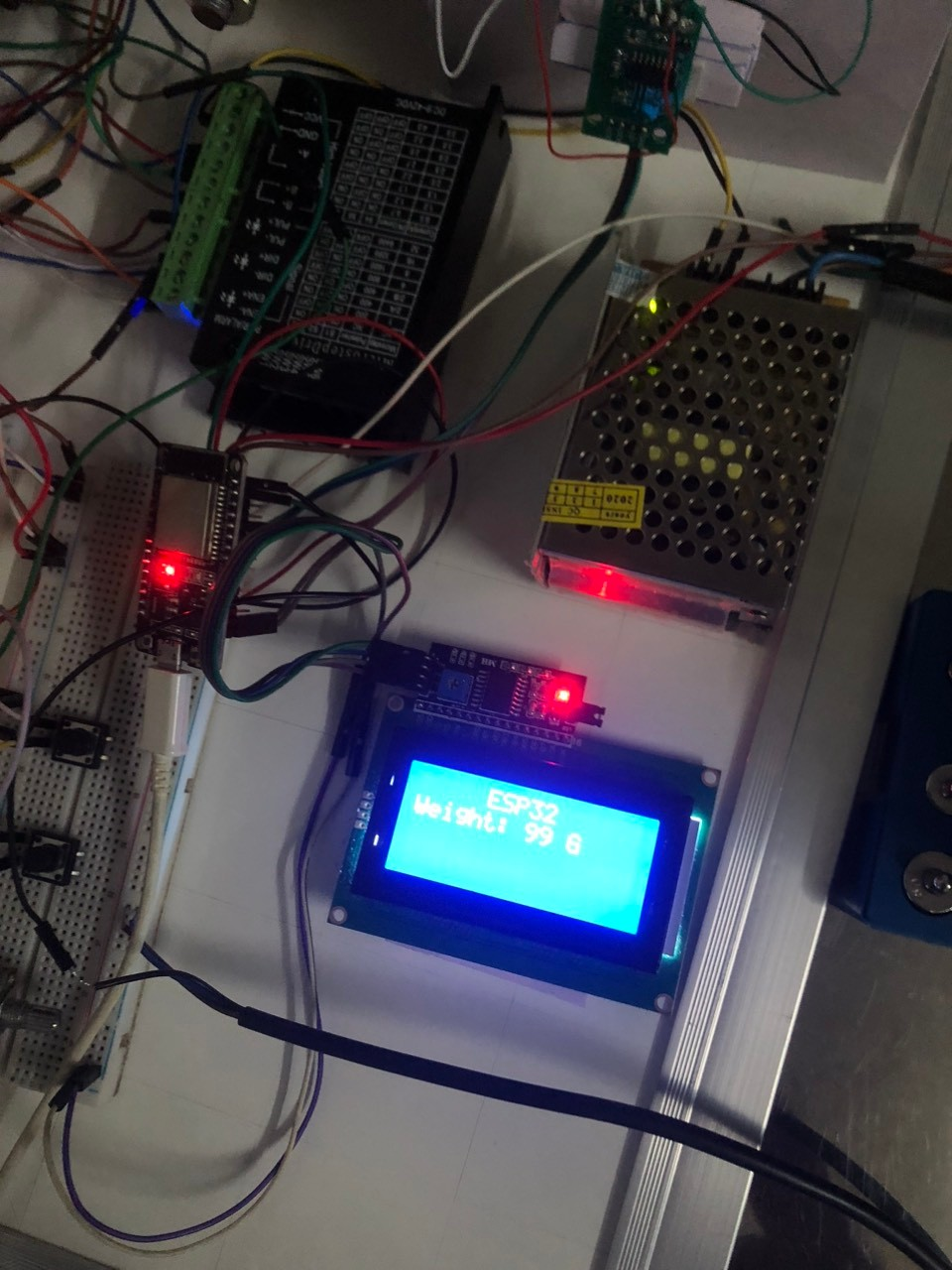
Vi điều khiển sử dụng chân A0 kết nối với một biến trở 10K Ohm. Hàm speedUP() sẽ trả về giá trị đọc được từ kênh A0 và gán cho biến customDelayMapped. Chương trình điều khiển động cơ bước bằng các lệnh cấp xung digitalWrite(stepPin, HIGH), digitalWrite(stepPin, LOW) thông qua module driver TB6600. Khi xoay biến trở, giá trị delay sẽ thay đổi giúp điều khiển được tốc độ quay của băng tải.

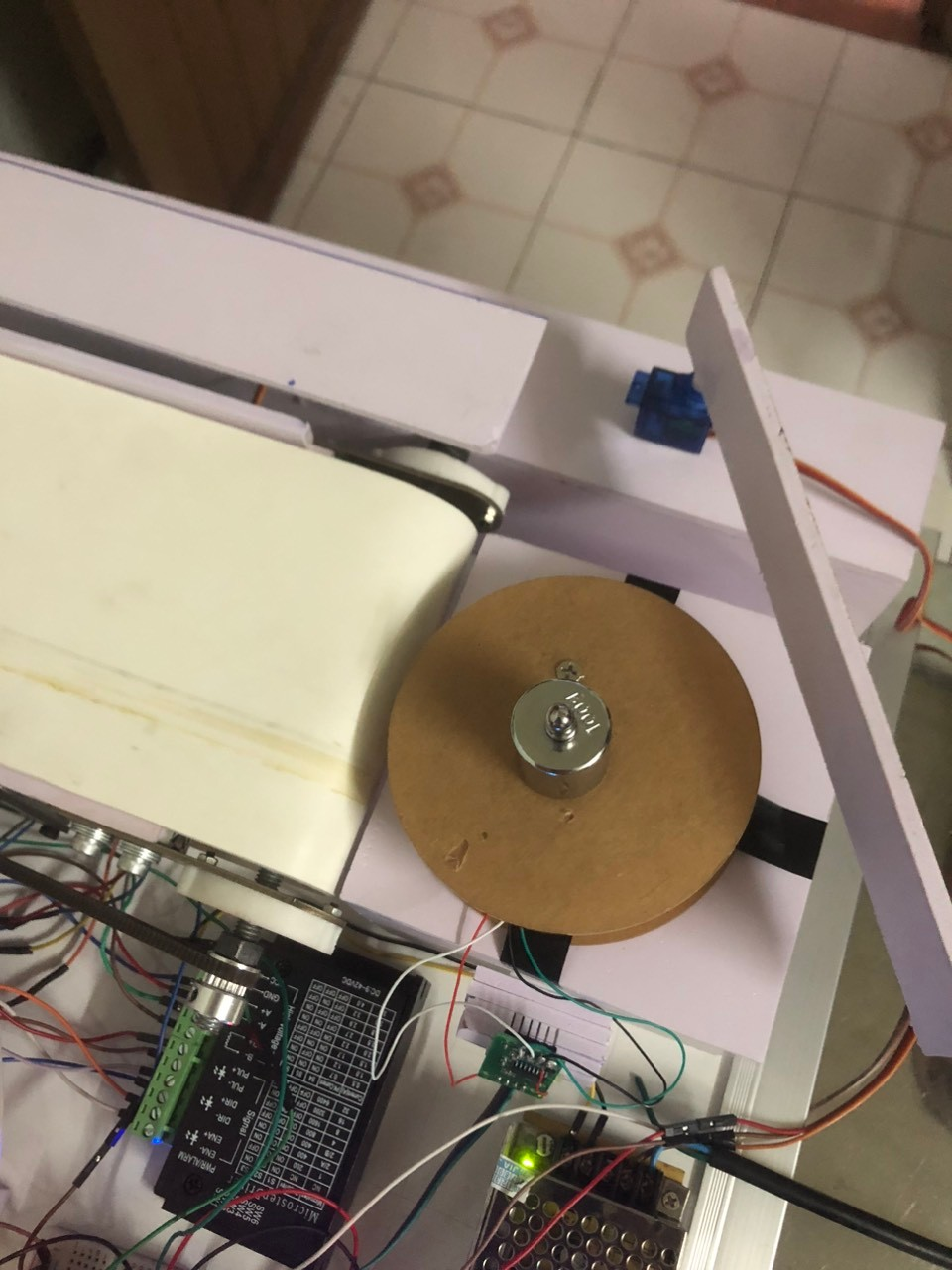
# kết quả

## KẾT QUẢ MÔ HÌNH THI CÔNG



## hoẠT ĐỘNG CỦA HỆ THỐNG





# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## KẾT LUẬN

Ưu điểm: thiết kế nhỏ gọn, cập nhật giá trị lên firebase, web nhanh và chính xác

Kết quả nhóm đạt được khá tốt với mục tiêu ban đầu đề ra, tuy nhiên độ chính xác của cảm biến còn hạn chế

Nhìn chung nhóm đã hoàn thành được các mục tiêu đề ra: điều chỉnh tốc độ băng tải, đo được cân nặng , phân loại sản phẩm chính xác theo tiêu chuẩn từng khoảng cân nặng, đếm số lượng sản phẩm , hiển thị và cập nhật lên lcd, firebase, web

## HƯỚNG PHÁT TRIỂN

Trong tương lai, hệ thống sẽ được phát triển theo hướng khắc phục những hạn chế hiện tại nêu trên. Bên cạnh đó sẽ bổ sung thêm nhiều chức năng hơn nữa như phân loại theo màu sắc, theo chiều cao,.. để tăng tính hiệu quả

TÀI LIỆU THAM KHẢO

1. Hướng dẫn điều khiển động cơ bước với thư viện Accel Stepper và driver điều khiển động cơ bước A4988 hoặc DRV8825
2. ESP32 with Load Cell and HX711 Amplifier (Digital Scale)
3. ESP32 Servo Motor Web Server with Arduino IDE
4. How to Use I2C LCD with ESP32 on Arduino IDE (ESP8266 compatible)