

**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN ĐHQG HCM**  
**KHOA ĐIỆN TỬ VIỄN THÔNG**



**BÁO CÁO ĐỒ ÁN**

**Đề tài: Hệ thống gạt nước ô tô tự động bằng cảm biến mưa dùng Arduino**

<b>Họ tên SV:</b>	<b>Lê Tuấn Kiệt</b>
	<b>Nguyễn Tuấn Kiệt</b>
<b>MSSV:</b>	<b>20200239</b>
	<b>20200240</b>
<b>Môn:</b>	<b>Cảm biến, đo và thiết bị đo</b>
	<b>nâng cao</b>
<b>Giảng viên:</b>	<b>Thầy Lê Việt Dũng</b>

**Năm học 2023-2024**

## LỜI MỞ ĐẦU

Với sự phát triển ngày càng nhanh chóng của khoa học kỹ thuật, việc sử dụng các vi điều khiển vào các thiết bị trong cuộc sống đang ngày càng trở nên phổ biến. Chỉ với một vài bộ vi điều khiển với những chip vi mạch nhỏ, chúng ta đã có thể sử dụng chúng điều khiển những hoạt động phức tạp hàng ngày một cách dễ dàng.

Không chỉ giúp chúng ta trong việc điều khiển các hoạt động hàng ngày, các bộ vi điều khiển còn giúp chúng đồng bộ hóa và chia sẻ thông tin. Với các lý do trên, việc tìm hiểu và nghiên cứu về vi điều khiển là điều mà các sinh viên kỹ thuật đặc biệt là sinh viên chuyên ngành Điện tử cần phải tìm hiểu và không ngừng phát triển. Đề tài này được thực hiện nhằm đáp ứng nhu cầu cần thiết và cấp bách đó của mỗi sinh viên.

Sau một thời gian trau dồi và tích lũy kiến thức từ các thầy cô trong khoa, đồng thời được sự giúp đỡ nhiệt tình của thầy Lê Việt Dũng và các bạn ở trong khoa, chúng em đã thiết kế và chế tạo “Hệ thống gạt nước ô tô tự động bằng cảm biến mưa dùng Arduino”.

Do thời gian, kiến thức cũng như trải nghiệm thực tế của chúng em còn hạn chế nên khó tránh khỏi những sai sót. Chúng em rất mong nhận được sự giúp đỡ và đóng góp ý kiến của thầy cô và các bạn để giúp cho đề tài phát triển hơn.

## MỤC LỤC

<b>Chương I: TỔNG QUAN ĐỀ TÀI</b> .....	5
1. Lý do chọn đề tài: .....	5
2. Giới hạn đề tài: .....	5
3. Nguồn tài liệu:.....	5
4. Nhiệm vụ nghiên cứu:.....	5
<b>Chương II: TỔNG QUAN CÁC LINH KIỆN CHÍNH</b> .....	6
1. ArduinoUno R3.....	6
1.1 Giới thiệu: .....	6
1.2 Uno: .....	7
1.3 Cấu tạo, thông số .....	8
2. Rain Sensor (Cảm biến mưa) .....	13
2.1 Giới thiệu: .....	13
2.2 Cấu tạo: .....	14
2.3 Nguyên lí hoạt động:.....	15
2.4 Thông số:.....	15
2.5 Ưu điểm:.....	16
2.6 Ứng dụng: .....	16
3. Servo Motor .....	16
3.1 Giới thiệu: .....	16
3.2 Cấu tạo: .....	17
3.3 Nguyên lí hoạt động:.....	18
3.4 Ưu nhược điểm của động cơ servo: .....	19
3.5 Ứng dụng: .....	19
<b>Chương III: HỆ THỐNG GẠT NƯỚC Ô TÔ TỰ ĐỘNG BẰNG CẢM BIẾN MƯA DÙNG ARDUINO UNO</b> .....	20

<b>1. Giới thiệu:</b>	20
<b>2. Các phần mềm thiết kế:</b>	20
<b>2.1 Arduino</b>	20
<b>2.2 Protues 8 Professional</b>	21
<b>3. Sơ đồ khối:</b>	22
<b>4. Nguyên lí hoạt động:</b>	22
<b>5. Lưu đồ thuật toán:</b>	24
<b>6. Code chương trình:</b>	25
<b>7. Mô phỏng:</b>	27
<b>8. Mạch thực tế:</b>	28
<b>Chương IV: KẾT LUẬN</b>	34
<b>1. Ưu/Nhược điểm:</b>	34
<b>2. Hướng phát triển:</b>	34
<b>TÀI LIỆU THAM KHẢO</b>	35

## **Chương I: TỔNG QUAN ĐỀ TÀI**

### **1. Lý do chọn đề tài:**

Với xu thế phát triển của xã hội, khoa học kỹ thuật nói chung và vi điều khiển nói riêng ngày càng được ứng dụng nhiều trong các lĩnh vực. Một trong những vi điều khiển phổ biến nhất hiện nay đó chính là Arduino. Việc tìm hiểu, khám phá, ứng dụng Arduino cùng các cảm biến vào trong hoạt động sản xuất và đời sống hàng ngày là một quá trình dài lí thú và hữu ích cho mỗi sinh viên. Trước thực tiễn này, chúng em đã quyết định chọn đề tài “Hệ thống gạt nước ô tô tự động bằng cảm biến mưa dùng Arduino”.

### **2. Giới hạn đề tài:**

Với những kiến thức thức đã được học và những gì đã tìm hiểu được trong thực tế, trong phạm vi của đồ án này, chúng em xin được trình bày sơ lược về cấu tạo, nguyên lí hoạt động của “Hệ thống gạt nước ô tô tự động bằng cảm biến mưa dùng Arduino”.

### **3. Nguồn tài liệu:**

Dựa vào mục đích tìm hiểu, phạm vi giới hạn và đối tượng nghiên cứu; trong quá trình thực hiện đề tài chúng em đã tham khảo những nguồn tài liệu sau:

- Giáo trình của các học phần: Điện tử căn bản, Điện tử số, Điện tử tương tự, Nhập môn kỹ thuật, Vi điều khiển, Đo và thiết bị đo, Kỹ thuật lập trình ngành Điện tử - Viễn thông, Cảm biến và thiết bị đo nâng cao.
- Các tài liệu về lập trình cho Arduino và mã hóa vi điều khiển.

### **4. Nhiệm vụ nghiên cứu:**

Thông qua quá trình nghiên cứu và thực hiện đề tài sinh viên có thể tìm hiểu và phát triển khả năng sử dụng vi điều khiển bằng cách:

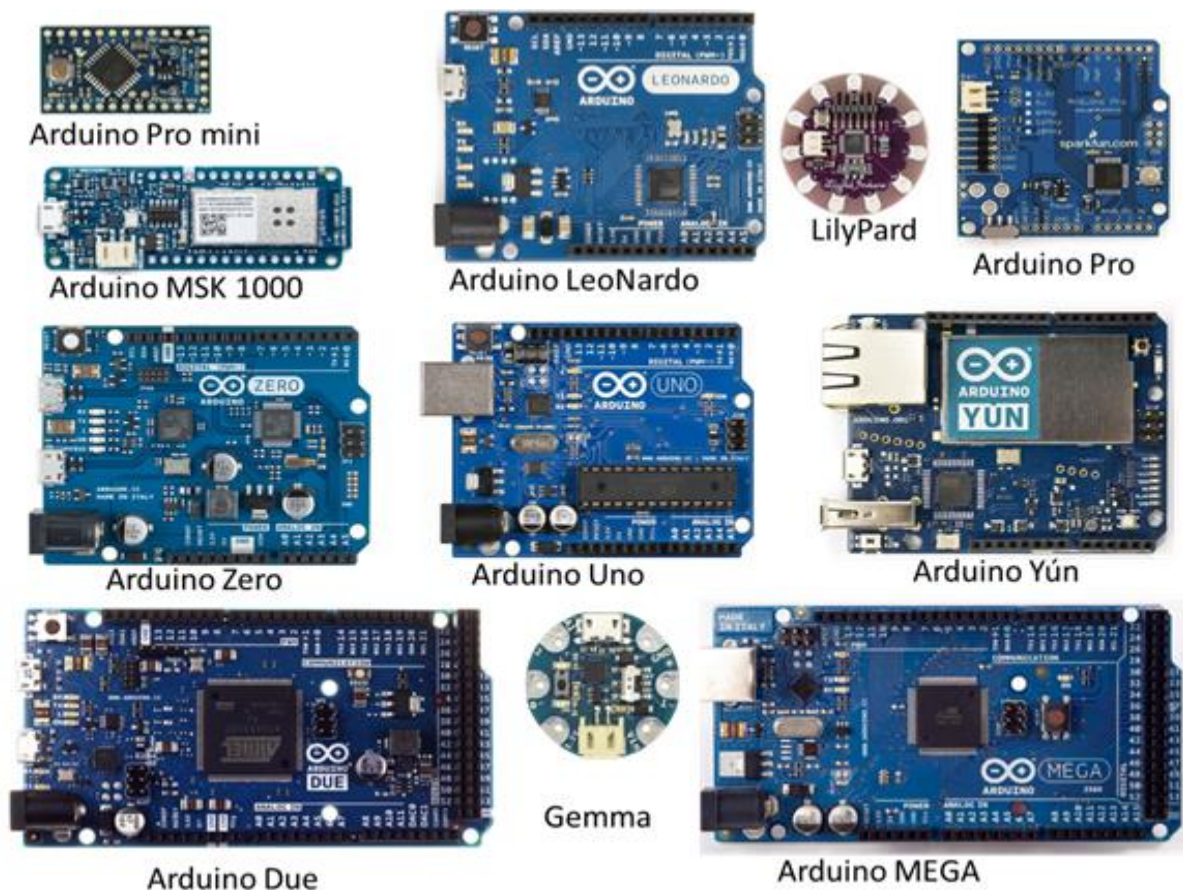
- Tìm hiểu cơ chế hoạt động.
- Phân tích sơ đồ nguyên lí.
- Nâng cao khả năng lập trình.
- Phát triển tư duy thiết kế hệ thống.

## Chương II: TỔNG QUAN CÁC LINH KIỆN CHÍNH

### 1. ArduinoUno R3

#### 1.1 Giới thiệu:

Arduino được tạo ra vào năm 2003 bởi một nhóm các giảng viên tại Viện Thiết kế Tương tác Ivrea ở Ý. Nó được đặt tên theo một quán bar ở cùng thị trấn Ivrea, nơi các giảng viên đã gặp nhau để thảo luận về dự án. Ban đầu, Arduino được phát triển như một nền tảng phần cứng cho một luận án của một sinh viên người Colombia tại Viện Thiết kế Tương tác Ivrea. Luận án này được đặt tên là “Arduino – La rivouzione dell’open hardware”(“Arduino – Cuộc cách mạng của phần cứng mã nguồn mở”). Sau đó, một nhóm gồm năm người đã tiếp tục phát triển nền tảng này và khi phiên bản mới hoàn thành, họ đã làm cho nó nhẹ hơn, giá rẻ hơn và có sẵn cho cộng đồng mã nguồn mở. Hiện nay, Arduino là một trong những phong trào phần cứng mã nguồn mở mạnh nhất trên thế giới.

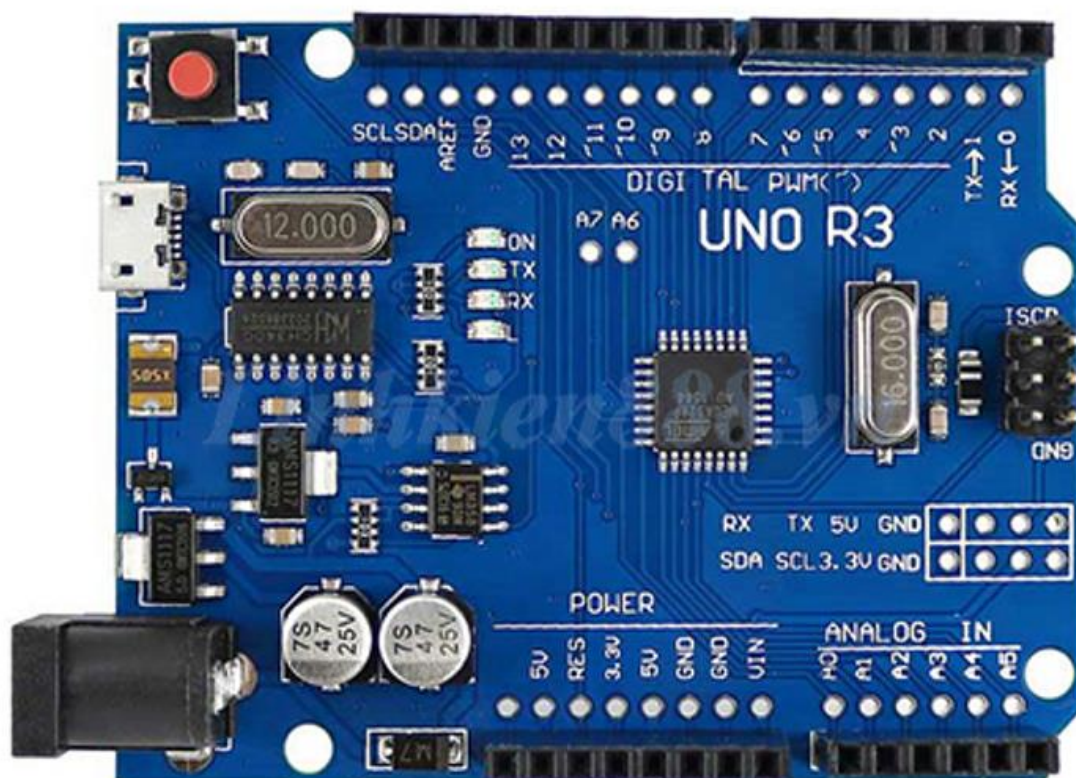


Hình 1.1: Những biến thể của Arduino

## 1.2 Uno:

“Uno” có nghĩa là một trong tiếng Ý và được đặt tên để đánh dấu việc phát hành của Arduino 1.0. và phiên bản 1.0 là phiên bản tài liệu tham khảo của Arduino.

Arduino Uno là một board điều khiển dựa trên Atmega328P. Nó có 14 chân đầu vào/ra kỹ thuật số (trong đó 6 chân có thể được sử dụng làm đầu ra PWM), 6 chân đầu vào tương tự, một tần số dao động gồm 16 MHz, kết nối USB một jack nguồn, một tiêu đề ICSP và một nút reset. Nó là board phổ biến nhất và được tài liệu hóa nhất trong toàn bộ gia đình Arduino. Để sử dụng nó chúng ta chỉ cần kết nối nó với máy tính bằng cáp USB hoặc cung cấp nguồn cho nó bằng adapter AC-to-DC hoặc pin để bắt đầu.



Hình 1.2: Arduino Uno R3



Uno khác với tất cả các phiên bản trước ở chỗ nó không sử dụng các FTDI chip điều khiển USB-to-serial. Thay vào đó, nó có tính năng Atmega 16U2 lập trình như là một công cụ chuyển đổi từ USB-to-serial.

Phiên bản 2 (R2) của Uno sử dụng Atmega8U2 có một điện trở kéo dòng 8U2 HWB xuống đất, làm cho nó dễ dàng hơn để đưa vào chế độ DFU.

Phiên bản 3 (R3) của Uno có các tính năng sau đây:

- Thêm SDA và SCL gần với pin Aref và hai chân mới được đặt gần với pin RESET, các IOREF cho phép thích ứng với điện áp cung cấp.
- Đặt lại mạch khỏe mạnh hơn.
- Atmega 16U2 thay thế 8U2.

### 1.3 Cấu tạo, thông số

#### 1.3.1 Cấu tạo:

Một board Arduino đời đầu gồm một cổng giao tiếp RS-232 (góc phía trên- bên trái) và một chip Atmel ATmega8 (màu đen, nằm góc phải-phía dưới); 14 chân I/O số nằm ở phía trên và 6 chân analog đầu vào ở phía đáy.

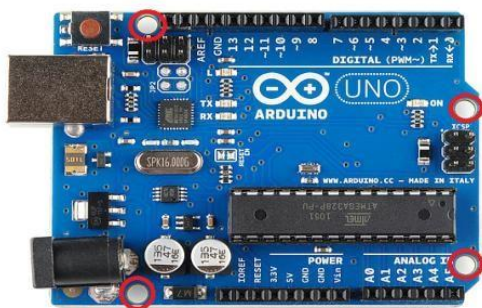


Hình 1.3: Arduino đời đầu

Board Arduino sẽ đưa ra hầu hết các chân I/O của vi điều khiển để sử dụng cho những mạch ngoài. Diecimila, Duemilanove, và bây giờ là Uno đưa ra 14 chân I/O kỹ thuật số, 6 trong số đó có thể tạo xung PWM (điều chế độ rộng xung) và 6 chân input analog, có thể được sử dụng như là 6 chân I/O số. Những chân này được thiết kế nằm phía trên mặt board, thông qua các header cái 0.10-inch (2.5 mm). Các board Arduino Nano, và Arduino-compatible Bare Bones Board và Boarduino có thể cung cấp các chân header được ở mặt trên của board dùng để cắm vào các breadboard.



Chiều dài tối đa và chiều rộng của Uno PCB là 2,7 và 2,1 inch tương ứng, với kết nối USB và jack điện mở rộng vượt ra ngoài không gian cũ. Bốn lỗ vít cho phép được gắn vào một bề mặt khác:



Hình 1.4: Các lỗ vít giúp cố định vị trí của Arduino

### 1.3.2 Vị trí và chức năng của các chân:

#### ❖ Các chân năng lượng

✚ **GND (Ground):** cực âm của nguồn điện cấp cho Arduino Uno. Khi bạn dùng các thiết bị sử dụng những nguồn điện riêng biệt thì những chân này phải được nối với nhau.

✚ **5V:** cấp điện áp 5V đầu ra. Dòng tối đa cho phép ở chân này là 500mA.

✚ **3.3V:** cấp điện áp 3.3V đầu ra. Dòng tối đa cho phép ở chân này là 50mA.

✚ **Vin (Voltage Input):** để cấp nguồn ngoài cho Arduino Uno, nối cực dương của nguồn với chân này và cực âm của nguồn với chân GND.

✚ **IOREF:** điện áp hoạt động của vi điều khiển trên Arduino Uno có thể được đo ở chân này. Và dĩ nhiên nó luôn là 5V. Mặc dù vậy không được lấy nguồn 5V từ chân này để sử dụng với chức năng của nó không phải là cấp nguồn.


✚ **RESET:** việc nhấn nút Reset trên board để reset vi điều khiển tương đương với việc chân RESET được nối với GND qua 1 điện trở 10KΩ.


*Lưu ý:*


\* Arduino Uno không có bảo vệ cắm ngược nguồn vào. Do đó bạn phải hết sức cẩn thận, kiểm tra các cực âm – dương của nguồn trước khi cấp cho Arduino Uno. Việc làm chập mạch nguồn vào của Arduino Uno sẽ biến nó thành một miếng nhựa chặn giấy vì vậy nên dùng nguồn từ cổng USB nếu có thể.

- \* Các chân 3.3V và 5V trên Arduino là các chân dùng để cấp nguồn ra cho các thiết bị khác, không phải là các chân cấp nguồn vào. Việc cấp nguồn sai vị trí có thể làm hỏng board. Điều này không được nhà sản xuất khuyến khích.
- \* Cấp nguồn ngoài không qua cổng USB cho Arduino Uno với điện áp dưới 6V có thể làm hỏng board.
- \* Cấp điện áp trên 13V vào chân RESET trên board có thể làm hỏng vi điều khiển Atmega328.
- \* Cường độ dòng điện vào/ra ở tất cả các chân Digital và Analog của Arduino Uno nếu vượt quá 200mA sẽ làm hỏng vi điều khiển.
- \* Cấp điện áp trên 5.5V vào các chân Digital hoặc Analog của Arduino Uno sẽ làm hỏng vi điều khiển.
- \* Cường độ dòng điện qua một chân Digital hoặc Analog bất kì của Arduino Uno vượt quá 40mA sẽ làm hỏng vi điều khiển. Do đó nếu không dùng để truyền nhận dữ liệu, bạn phải mắc một điện trở hạn dòng.

❖ Bộ nhớ

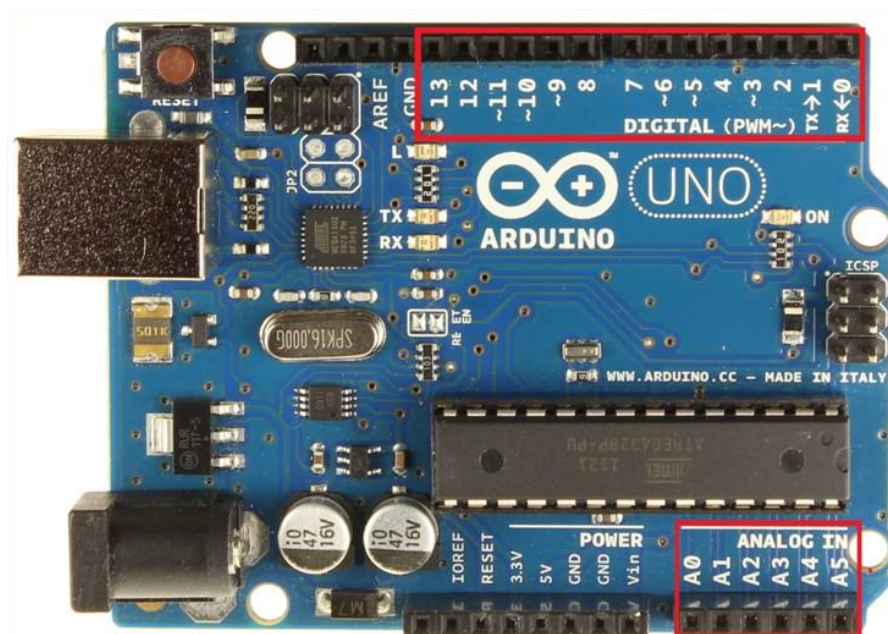
 **32KB bộ nhớ Flash:** những đoạn lệnh lập trình sẽ được lưu trữ trong bộ nhớ Flash của vi điều khiển.

 **2KB cho SRAM (Static Random Access Memory):** giá trị các biến khai báo khi lập trình sẽ lưu ở đây, khai báo càng nhiều biến thì càng cần nhiều bộ nhớ RAM. Khi mất điện, dữ liệu trên SRAM sẽ bị mất.

 **1KB cho EEPROM (Electrically Erasable Programmable Read Only Memory):** đây giống như một chiếc ổ cứng mini – nơi có thể đọc và ghi dữ liệu vào mà không phải lo bị mất khi cúp điện giống như dữ liệu trên SRAM.

❖ Các cổng vào/ra

Arduino Uno có 14 chân digital dùng để đọc hoặc xuất tín hiệu. Chúng chỉ có 2 mức điện áp là 0V và 5V với dòng vào/ra tối đa trên mỗi chân là 40mA. Ở mỗi chân đều có các điện trở pull-up từ được cài đặt ngay trong vi điều khiển ATmega328 (mặc định thì các điện trở này không được kết nối).



Hình 1.5: Các chân của Arduino Uno

Một số chân digital có chức năng đặc biệt như sau:

✚ **2 chân Serial:** 0 (RX) và 1 (TX): dùng để gửi (transmit – TX) và nhận (receive – RX) dữ liệu TTL Serial. Arduino Uno có thể giao tiếp với thiết bị khác thông qua 2 chân này. Kết nối bluetooth thường thấy nói nôm na chính là kết nối Serial không dây. Nếu không cần giao tiếp Serial, không nên sử dụng 2 chân này nếu không cần thiết.

✚ **Chân PWM (~): 3, 5, 6, 9, 10 và 11:** cho phép bạn xuất ra xung PWM với độ phân giải 8bit (giá trị từ  $0 \rightarrow 2^8 - 1$  tương ứng với  $0 \rightarrow 5V$ ) bằng hàm `analogWrite()`. Nói một cách đơn giản, có thể điều chỉnh được điện áp ra ở chân này từ mức 0V đến 5V thay vì chỉ cố định ở mức 0V và 5V như những chân khác.

✚ **Chân giao tiếp SPI:** 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Ngoài các chức năng thông thường, 4 chân này còn dùng để truyền phát dữ liệu bằng giao thức SPI với các thiết bị khác.

✚ **LED 13:** trên Arduino Uno có 1 đèn led màu cam (kí hiệu chữ L). Khi bấm nút Reset, bạn sẽ thấy đèn này nhấp nháy để báo hiệu. Nó được nối với chân số 13. Khi chân này được sử dụng, LED sẽ sáng.

Arduino Uno có 6 chân analog (A0  $\rightarrow$  A5) cung cấp độ phân giải tín hiệu 10bit ( $0 \rightarrow 2^{10} - 1$ ) để đọc giá trị điện áp trong khoảng 0V  $\rightarrow$  5V. Với chân

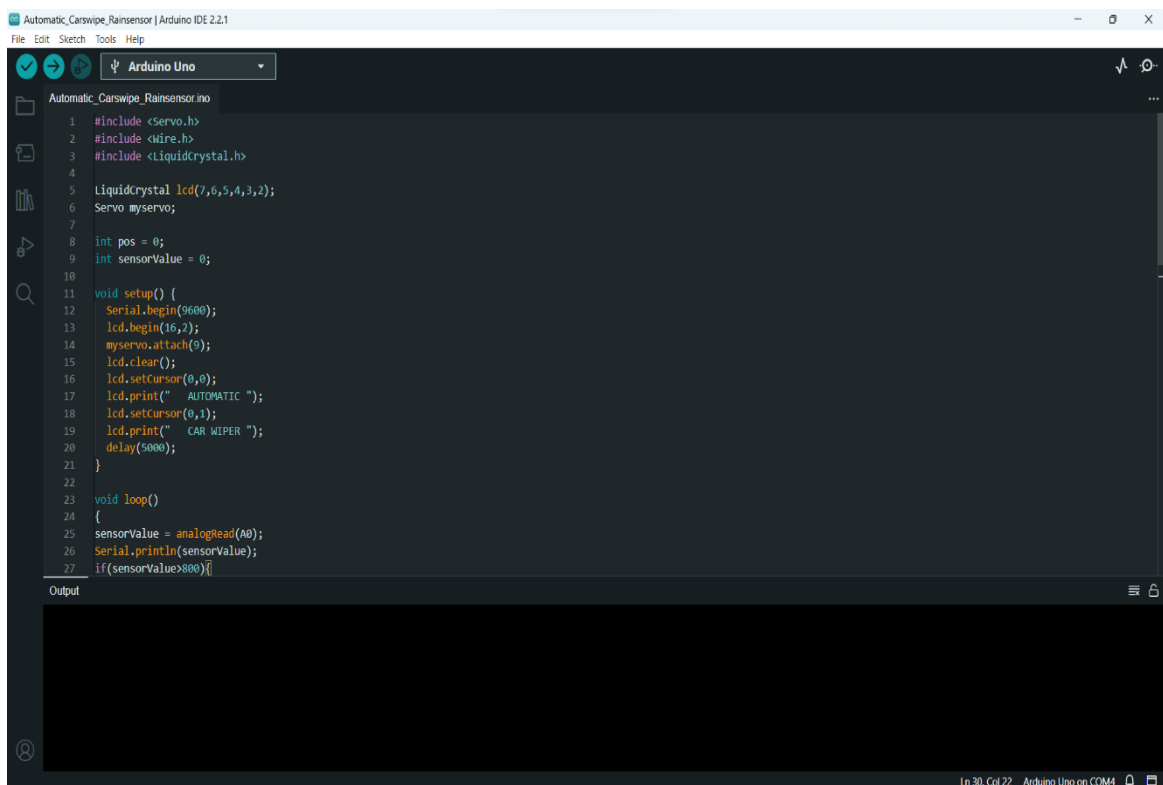
**AREF** trên board, bạn có thể đưa vào điện áp tham chiếu khi sử dụng các chân analog. Tức là nếu cấp điện áp 2.5V vào chân này thì có thể dùng các chân analog để đo điện áp trong khoảng từ 0V → 2.5V với độ phân giải vẫn là 10bit.

Đặc biệt, Arduino Uno có 2 chân A4 (SDA) và A5 (SCL) hỗ trợ giao tiếp I2C/TWI các thiết bị khác.

### ❖ Lập trình cho Arduino

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 nói chung. Và Wiring lại là một biến thể của C/C++. Ngôn ngữ Arduino bắt nguồn từ C/C++ phổ biến hiện nay rất dễ học, dễ hiểu.

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 được gọi là Arduino IDE (Intergrated Development Enviroment) như hình dưới đây.



Hình 1.6: Chương trình minh họa

### 1.3.3 Thông số:

Một vài thông số của Arduino Uno R3:

Vi điều khiển	ATmega328 họ 8bit
Điện áp hoạt động	5V DC (chỉ được cấp qua cổng USB)
Tần số hoạt động	16 MHz
Dòng tiêu thụ	khoảng 30mA
Điện áp vào khuyến dùng	7-12V DC
Điện áp vào giới hạn	6-20V DC
Số chân Digital I/O	14 (6 chân hardware PWM)
Số chân Analog	6 (độ phân giải 10bit)
Dòng tối đa trên mỗi chân I/O	30 mA
Dòng ra tối đa (5V)	500 mA
Dòng ra tối đa (3.3V)	50 mA
Bộ nhớ flash	32 KB (ATmega328) với 0.5KB dùng bởi bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)

## 2. Rain Sensor (Cảm biến mưa)

### 2.1 Giới thiệu:

Cảm biến mưa (Rain water sensor) là một thiết bị chuyên dụng có khả năng nhận biết được giọt nước, trời mưa, lượng mưa hoặc môi trường có nước.

Cảm biến mưa hoạt động tương tự như một công tắc. Khi có hạt mưa rơi trên bề mặt của tấm cảm biến thì mô-đun cảm biến sẽ đọc dữ liệu từ tấm cảm

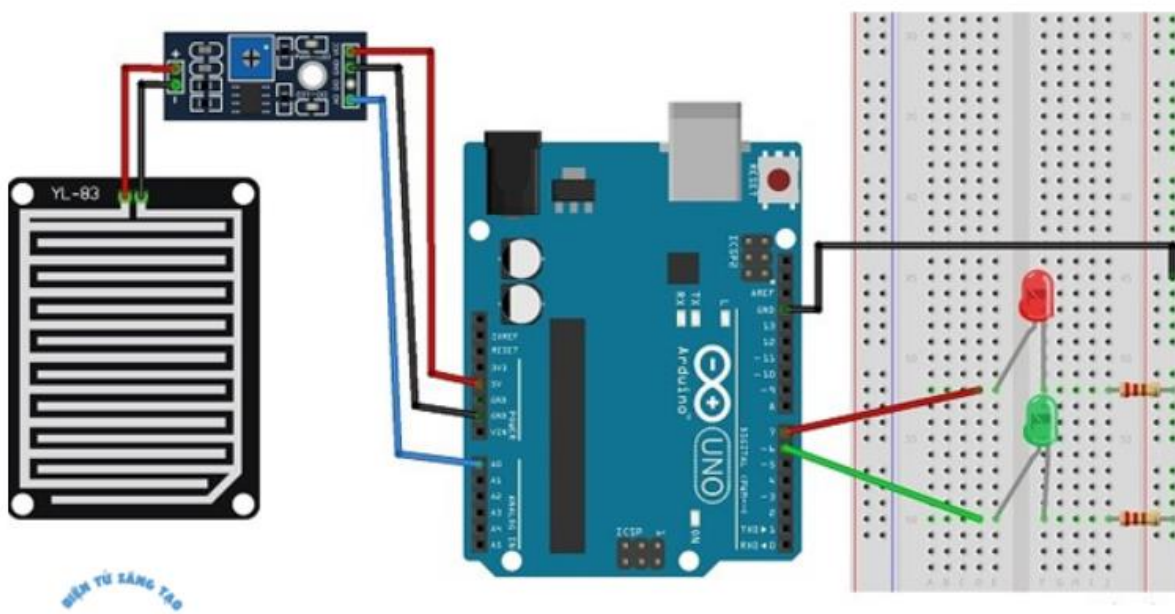
biến để xử lý và chuyển nó thành đầu ra kỹ thuật số hoặc tín hiệu tương tự. Tín hiệu đầu ra được truyền về sử dụng để điều khiển một thiết bị nào đó như mái che, còi báo, hay điều khiển motor, .. tùy theo mục đích của trường hợp.

Hiện nay, có nhiều loại cảm biến mưa được cung cấp trên thị trường nhưng đại đa số chỉ là bo mạch nhỏ Arduino. Bo mạch thường dùng để phục vụ cho thí nghiệm, nghiên cứu khoa học và chưa ứng dụng nhiều trong thực tiễn.

Cảm biến mưa còn được gọi là cảm biến báo nước mưa, cảm biến trời mưa, bộ báo có nước mưa, bộ báo trời mưa,...

## 2.2 Cấu tạo:

Cảm biến mưa có cấu tạo gồm hai bộ phận chính đó là đệm cảm biến và mô-đun cảm biến. Khi nước mưa rơi trên bề mặt của tấm cảm biến thì module cảm biến sẽ đọc dữ liệu để xử lý và chuyển thành đầu ra dạng tín hiệu tương tự (Analog) hoặc kỹ thuật số (Digital). Đó là lý do mà đầu ra cảm biến báo mưa sẽ dạng tín hiệu là tương tự (Analog-AO) và kỹ thuật số (Digital-DO).



Hình 2.1: Mô-đun cảm biến mưa

Đối với mạch cảm biến báo mưa sẽ gồm 2 bộ phận chính đó là:

- Bộ phận cảm biến mưa được gắn ngoài trời để tiếp xúc với hạt mưa.
- Bộ phận mạch lập trình và điều chỉnh độ nhạy cần được che chắn và đặt bên trong.



### 2.3 Nguyên lý hoạt động:

Cảm biến mưa hoạt động dựa theo cách so sánh hiệu điện thế của mạch cảm biến nằm ngoài trời với một giá trị định trước. Giá trị định trước này thay đổi được thông qua 1 biến trở màu xanh. Từ kết quả so sánh thì bộ xử lý sẽ phát ra tín hiệu đóng hoặc ngắt rơ le qua chân DO.

Khi xuất hiện nước trên bề mặt cảm biến (trời mưa), độ dẫn điện của tấm bên ngoài sẽ tốt hơn khiến cho điện trở giảm xuống. Chân DO sẽ được kéo xuống thấp (0V), đèn LED màu đỏ sẽ được bật sáng lên.

Khi trời không mưa, độ dẫn điện của vật liệu sẽ kém khiến cho điện trở cao, chân DO của module cảm biến mưa được giữ ở mức cao (khoảng từ 5V-12V). Vì vậy, đầu ra của cảm biến mưa chủ yếu phụ thuộc vào điện trở.

### 2.4 Thông số:

Các thông số kỹ thuật cơ bản của cảm biến mưa:

- Điện áp sử dụng: 5V
- Đèn báo hiệu nguồn và đầu ra: Đèn led báo nguồn có xanh, đèn led cảnh báo mưa có màu đỏ. LED sáng lên khi không có mưa đầu ra cao. Khi có mưa, đầu ra thấp nên đèn LED tắt.
- Nguyên lý hoạt động: Nước rơi vào board sẽ tạo ra môi trường dẫn điện làm giảm điện trở.
- Tín hiệu đầu ra:
  - ✚ Tín hiệu Digital ở chân  $D_0$  cho biết cảm biến ướt hay khô dựa trên một giá trị ngưỡng được thiết lập trước bằng biến trở. Khi cảm biến ướt, tín hiệu ngõ ra là 0V(mức thấp) và khi cảm biến khô, tín hiệu ngõ ra là 5V(mức cao).

✚ Tín hiệu Analog ở chân  $A_0$  được tính theo công thức:

$$V_{out} = V_{CC} \times \frac{R_2}{R_1 + R_2}$$

Trong đó:

$V_{CC}$  là điện áp cấp vào 5V.

$R_1$  là điện trở của cảm biến, điện trở này giảm khi cảm biến ướt và tăng khi nó khô trong khoảng từ 500Ω đến 10 KΩ.

$R_2$  là điện trở của biến trở trên mô-đun, ta có thể điều chỉnh để có thể tăng giảm độ nhạy trong khoảng từ 0 đến 2 KΩ.



- Độ nhạy có thể được điều chỉnh thông qua một chiết áp để có thể điều chỉnh cho phù hợp với nhu cầu sử dụng.

## **2.5 Ưu điểm:**

- Cảm biến siêu nhạy, đưa ra thông báo nhanh chóng khi phát hiện trời mưa.
- Dễ dàng đấu nối với các thiết bị điều khiển.
- Chi phí mua cảm biến mưa không cao nhưng hiệu quả mang lại rất tốt.
- Tuổi thọ cao, lắp đặt ngoài trời không cần hộp che.
- Tốn ít chi phí bảo hành, bảo dưỡng.
- Tính an toàn cao, an toàn với trẻ nhỏ và người già và dễ dàng lắp đặt ở vị trí phù hợp.

## **2.6 Ứng dụng:**

Trong thực tế, cảm biến báo mưa được dùng như một thiết bị bảo quản nước. Nó có thể được kết nối hệ thống tưới tiêu để tắt hệ thống trong trường hợp xuất hiện mưa.

Cảm biến mưa còn ứng dụng để làm thiết bị hỗ trợ tự động hóa trong kéo giá phơi quần áo, tự động đóng cửa khi trời mưa,...

Cảm biến này còn được ứng dụng để bảo vệ các thiết bị bên trong xe hơi chống lại lượng mưa, bật tắt tự động gạt nước kính chắn gió.

Cảm biến mưa còn ứng dụng trong các thiết bị vệ tinh thông tin liên lạc chuyên dụng để kích hoạt một máy thổi mưa khi mở cửa cấp liệu ở trên không, nhờ đó sẽ loại bỏ các giọt nước khỏi màng bọc mylar để giữ áp suất cũng như không khí khô trong ống dẫn sóng.

## **3. Servo Motor**

### **3.1 Giới thiệu:**

Động cơ điện servo hay còn gọi là servo motor là một loại động cơ máy móc chuyên dùng để cung cấp cơ năng cho một số thiết bị, dây chuyền hay cơ cấu hoạt động nào đó trong quy trình sản xuất và chế tạo. Chúng đóng vai trò là đầu tàu cung cấp lực kéo, giúp cho các dây chuyền hay các cơ cấu động cơ khác hoạt động theo.

Thiết bị này thường sử dụng từ trường để biến điện năng thành cơ năng dưới dạng xoay nhằm mục đích kéo tải. Động cơ được kết nối với các thiết bị, cơ cấu động cơ khác bằng hệ thống truyền, đai, xích hay bơm,...

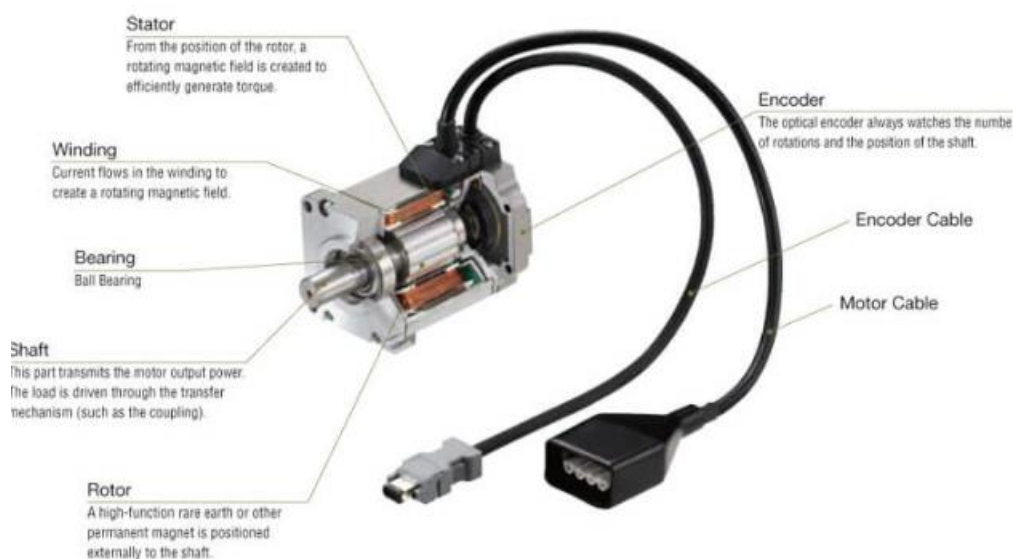


Hình 2.2: Các loại servo motor

### 3.2 Cấu tạo:

Động cơ servo là một loại động cơ điện có khả năng quay một góc chính xác đến một vị trí cụ thể. Để làm được nhiệm vụ này, động cơ servo có cấu tạo bao gồm:

- Stator: Stator là phần cố định của động cơ, được làm bằng một cuộn dây điện.
- Rotor: Rotor là phần quay của động cơ, được làm bằng một nam châm vĩnh cửu.
- Bộ phận điều khiển: Bộ phận điều khiển là bộ phận cung cấp điện cho động cơ và điều khiển tốc độ và vị trí của rotor.



Hình 2.3: Cấu tạo của servo motor

### 3.3 Nguyên lý hoạt động:

Động cơ servo hoạt động dựa trên nguyên lý từ trường như sau:

- Rotor của động cơ servo motor chính là một nam châm vĩnh cửu.
- Động cơ này có từ trường mạnh, đồng thời stator của động cơ còn được cuốn vào các cuộn dây riêng biệt, được cấp nguồn điện để hoạt động theo một trình tự thích hợp, từ đó sẽ làm quay rotor.
- Nếu thời điểm mà dòng điện cấp tới cho các cuộn dây là chuẩn xác thì khi đó chuyển động quay của roto sẽ phụ thuộc vào tần số và pha của dòng điện, mặt khác, phân cực và dòng điện sẽ chạy trong cuộn dây stator.
- Động cơ servo được tạo nên bởi những hệ thống hồi tiếp vòng khép kín và tín hiệu đầu ra của động cơ. Chúng sẽ được nối với một mạch điều khiển. Khi động cơ quay thì vận tốc và vị trí của chúng sẽ được hồi tiếp về mạch điều khiển này.
- Khi đó, cho dù bất kì tác nhân nào muốn ngăn cản chuyển động quay của động cơ thì cơ cấu hồi tiếp sẽ nhận được tín hiệu cho thấy chưa đạt được vị trí mong muốn. Mạch điều khiển sẽ tiếp tục chỉnh sai lệch, điều này khiến cho động cơ đạt được vị trí chính xác nhất của bộ điều khiển servo.

### 3.4 Ưu nhược điểm của động cơ servo:

#### 3.4.1 Ưu nhược điểm của động cơ servo AC:

✓ Ưu điểm:

- Điều khiển tốc độ cực tốt, điều khiển một cách trơn tru trên toàn bộ vùng tốc độ.
- Hiệu suất hoạt động cao hơn 90%.
- Động cơ sản sinh ít nhiệt, điều khiển được ở tốc độ cao.
- Điều khiển vị trí có độ chính xác cao (tùy vào độ chính xác tuyệt đối của bộ mã hóa).
- Động cơ dùng momen xoắn, quán tính thấp, đồng thời tiếng ồn thấp, không có bàn chải mặc, với chế độ bảo trì miễn phí áp dụng cho môi trường không có khói bụi, chất dễ gây nổ.

\* Nhược điểm:

- Điều khiển động cơ AC sẽ phức tạp hơn, vì các thông số ổ đĩa cũng cần phải được điều chỉnh theo các thông số PID chính xác để có thể xác định được nhu cầu kết nối.

#### 3.4.2 Ưu nhược điểm của động cơ servo DC:

✓ Ưu điểm:

- Kiểm soát được tốc độ chính xác cho máy móc.
- Nguyên tắc điều khiển tương đối đơn giản, dễ sử dụng.
- Giá rẻ hơn các loại khác.

\* Nhược điểm:

- Chổi than của động cơ sẽ giới hạn tốc độ, sức đề kháng bổ sung, do đó dẫn đến các hạt bị mài mòn (do môi trường không có bụi bản sẽ không thích hợp).

### 3.5 Ứng dụng:

Động cơ servo thường được ứng dụng trong lĩnh vực điện và điện tử. Khi máy móc được lắp ráp trong dây chuyền sản xuất đòi hỏi phải có tốc độ cao thì động cơ Servo đáp ứng được tốt nhất yêu cầu này. Cụ thể:

- Ứng dụng động cơ Servo trong điều khiển vận chuyển: Động cơ servo sẽ giúp điều khiển tốc độ nhanh hay chậm tùy theo mục đích sử dụng.
- Ứng dụng động cơ Servo về khuôn mẫu đúc trong lĩnh vực sản xuất nhựa.
- Ứng dụng động cơ Servo trong ngành điện – điện tử.
- Ứng dụng động cơ Servo trong ngành sản xuất thực phẩm

### Chương III: HỆ THỐNG GẠT NƯỚC Ô TÔ TỰ ĐỘNG BẰNG CẢM BIẾN MƯA DÙNG ARDUINO UNO

#### 1. Giới thiệu:

Đồ án “Hệ thống gạt nước ô tô tự động bằng cảm biến mưa dùng Arduino Uno” được thực hiện với mục đích ứng dụng các cảm biến và vi điều khiển vào cuộc sống hàng ngày. Từ đó, giúp mọi người nhận thấy được tiềm năng và sự phát triển của tự động hóa mang đến cho chúng ta rất nhiều tiện ích.

#### 2. Các phần mềm thiết kế:

##### 2.1 Arduino

Arduino là môi trường phát triển tích hợp mã nguồn mở, cho phép người dùng dễ dàng viết code và tải nó lên board mạch, được viết bằng Java dựa trên ngôn ngữ lập trình và phần mềm mã nguồn mở khác.



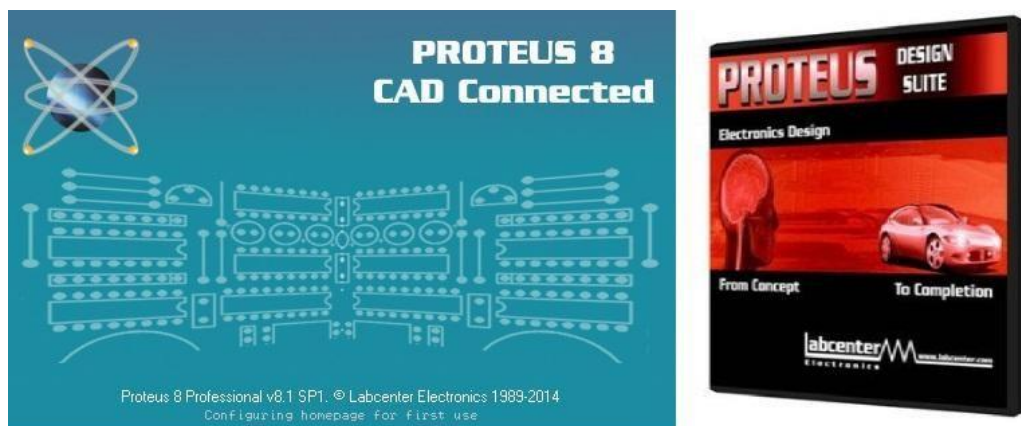
Hình 3.1: Arduino IDE

Kể từ tháng 3 năm 2015, Arduino IDE (Integrated Development Editor – môi trường phát triển thích hợp) đã được phổ biến tại rất nhiều nơi với giao diện trực quan.

Ngôn ngữ phổ quát cho Arduino là C và C++. Do đó phần mềm phù hợp với những người dùng quen thuộc các ngôn ngữ này.

Phần mềm gồm những mảng thư viện phong phú như: EEPROM, Firmata, GSM, Servo, TFT, Wifi,... Và các mảng thư viện ngày càng đa dạng nhờ sự đóng góp của cộng đồng Arduino trên toàn thế giới.

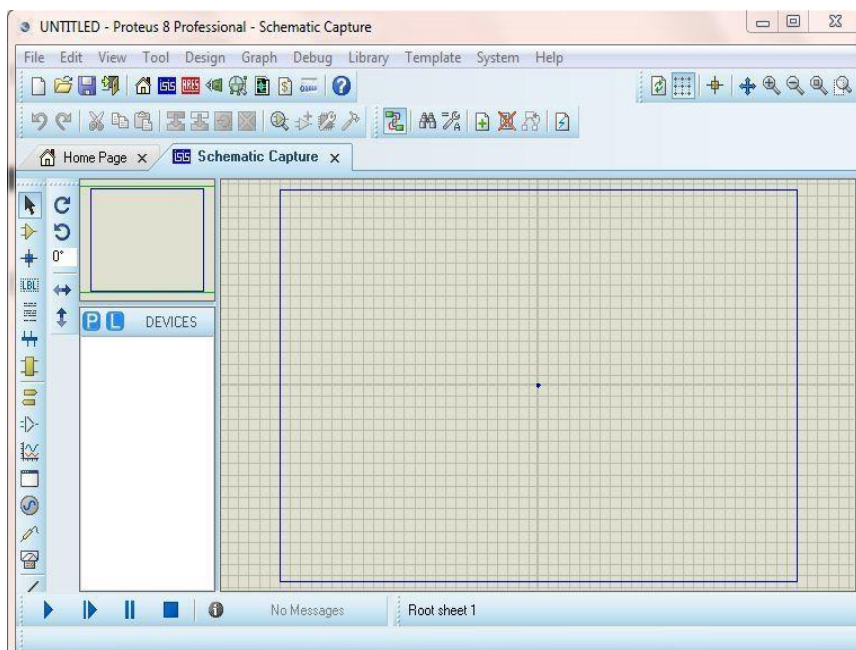
## 2.2 Protues 8 Professional



Hình 3.2: Protues Labcenter Electronics

Proteus là phần mềm mô phỏng mạch điện tử của Labcenter Electronics, mô phỏng cho hầu hết các linh kiện điện tử thông dụng, đặc biệt hỗ trợ cho cả các MCU như PIC, 8051, AVR, ...

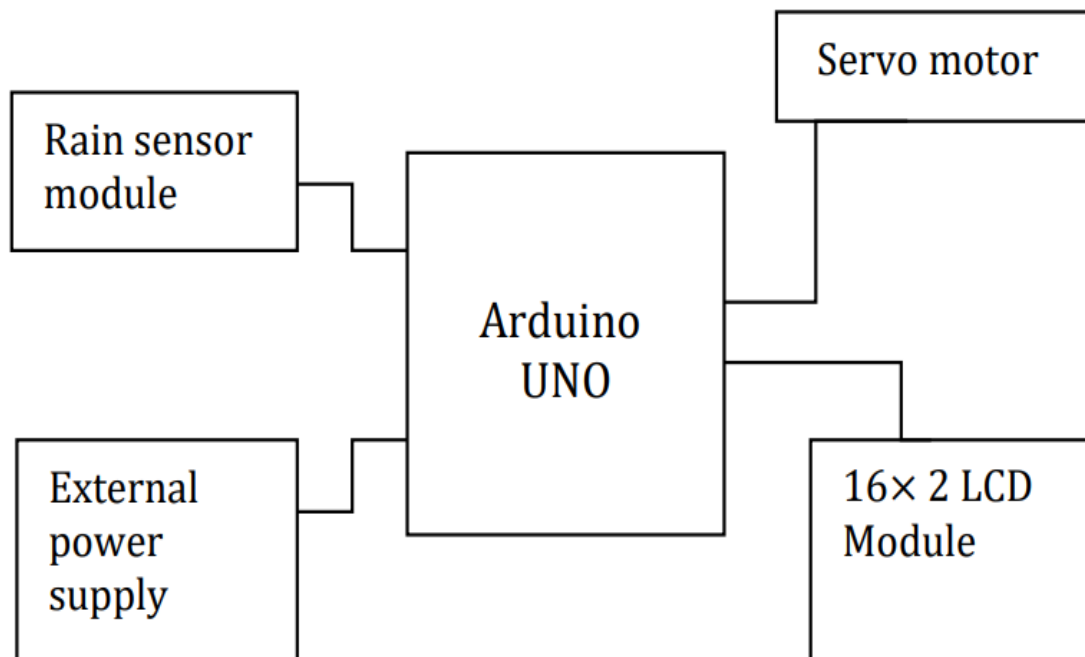
Phần mềm bao gồm 2 mảng chính là ISIS cho phép mô phỏng mạch điện tử và ARES dùng để vẽ mạch in.



Hình 3.3: Giao diện chính của Protues

### 3. Sơ đồ khối:

Hệ thống bao gồm một board Arduino Uno R3, một cảm biến mưa, động cơ servo và một mô-đun LCD. Sơ đồ khối của hệ thống xuất hiện ở hình 3.2. Hệ thống được chia ra làm 4 giai đoạn: giai đoạn đầu là giai đoạn đọc dữ liệu từ cảm biến mưa, giai đoạn thứ hai là giai đoạn xử lý thông tin từ cảm biến, giai đoạn thứ ba là giai đoạn so sánh và phân tích thông tin, giai đoạn cuối cùng là giai đoạn điều khiển động cơ và hiển thị lên LCD. Ngôn ngữ được sử dụng để lập trình là ngôn ngữ lập trình Arduino.



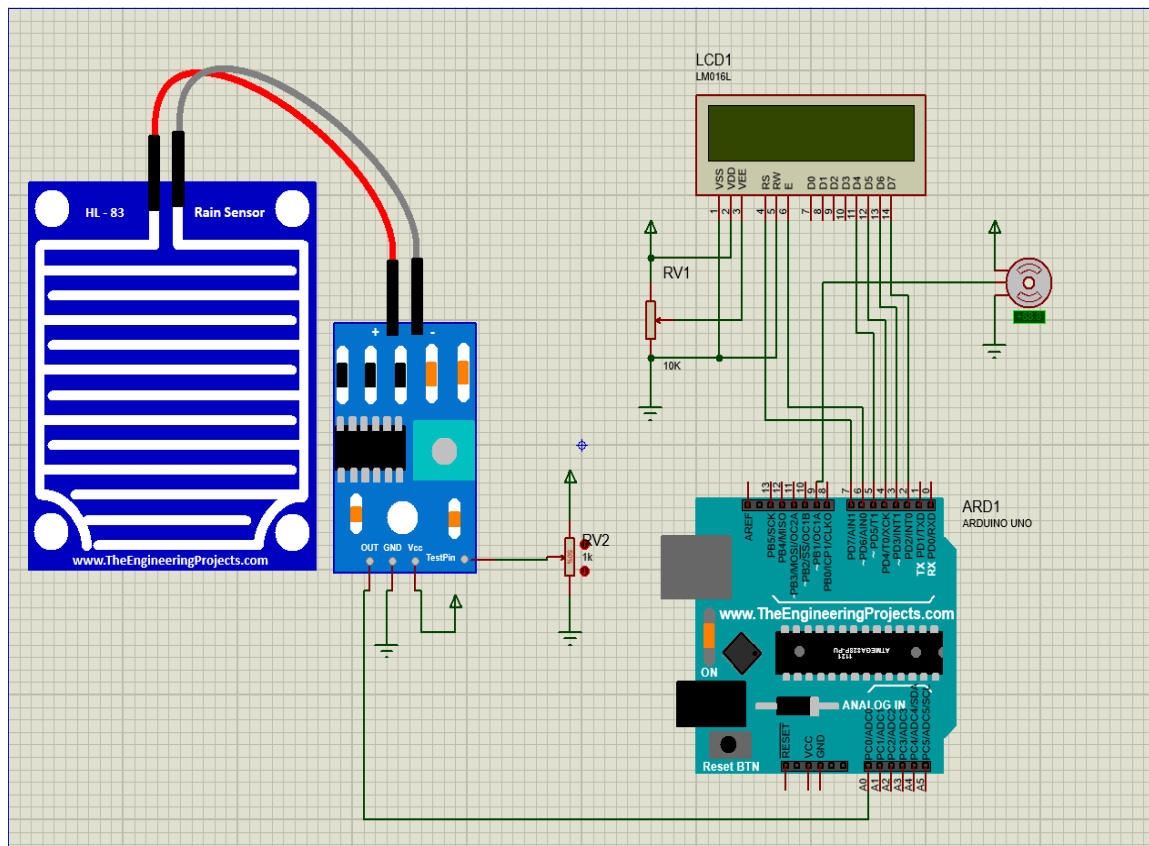
Hình 3.4: Sơ khối của hệ thống

### 4. Nguyên lí hoạt động:

Khi trời mưa, cảm biến mưa sẽ có nước và điện trở sẽ bị thay đổi. Vì vậy, cảm biến hoạt động như một biến trở. Mối quan hệ giữa cường độ mưa và điện trở tỷ lệ nghịch với nhau. Nếu như số lượng hạt mưa tăng lên thì điện trở của cảm biến giảm. Cảm biến sẽ gửi tín hiệu và tín hiệu được nhận bởi vi điều khiển. Bộ vi điều khiển xác định cường độ truyền và truyền tín hiệu đến

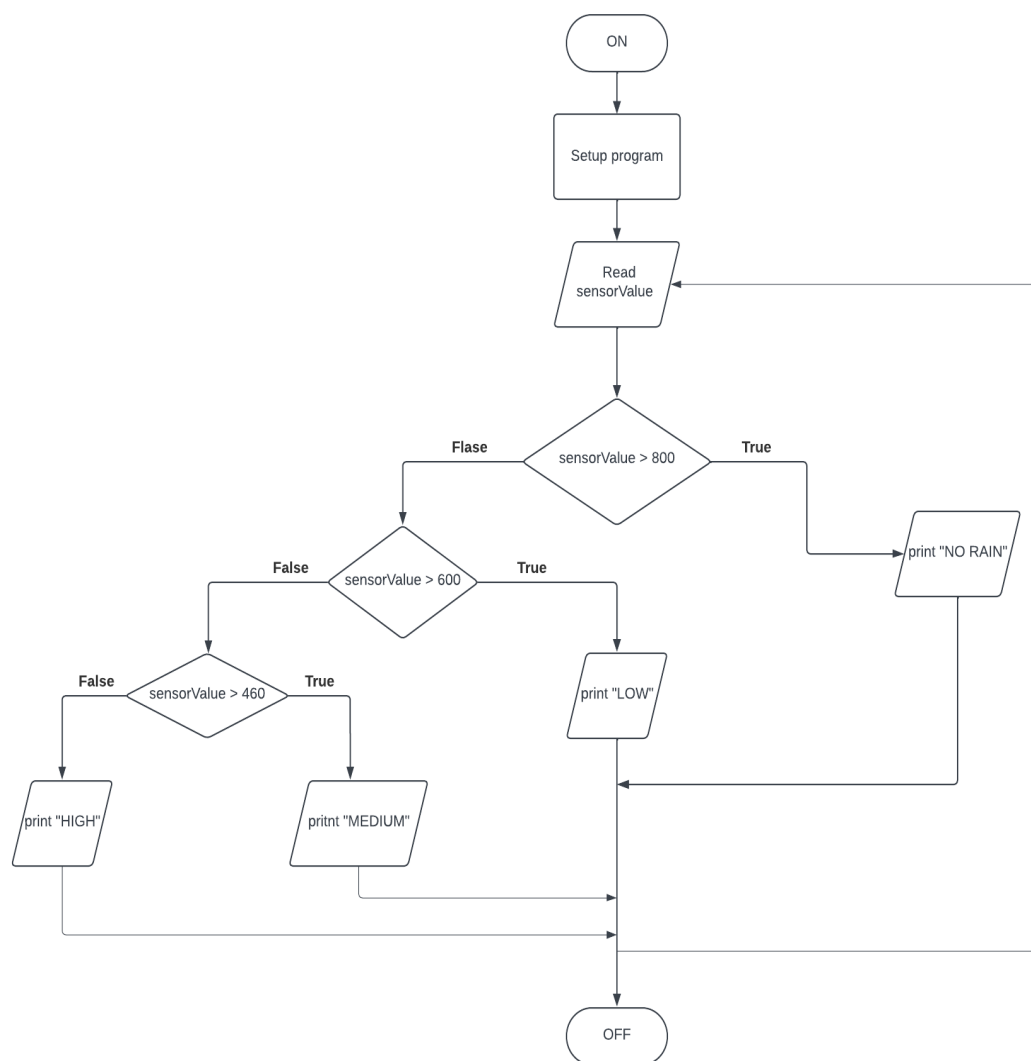


động cơ servo trong dạng điều chế độ rộng xung. Sau đó, cần gạt nước sẽ được hoạt động theo chế độ phù hợp dựa vào lượng mưa. Cảm biến được thiết kế với kích thước sao cho không làm cản trở tầm nhìn của người lái xe và có khả năng chống lại các yếu tố từ môi trường. Vì thế, cảm biến sẽ không gửi báo động giả. Sự sụt giảm điện trở được ghi lại như một tín hiệu để vi điều khiển xác định cường độ mưa. Tín hiệu được gửi đến mô tơ servo, mô tơ này vận hành các cần gạt nước. Tốc độ của cần gạt nước tăng lên khi sức mạnh tăng lên.



Hình 3.5: Sơ đồ nguyên lí hoạt động

## 5. Lưu đồ thuật toán:

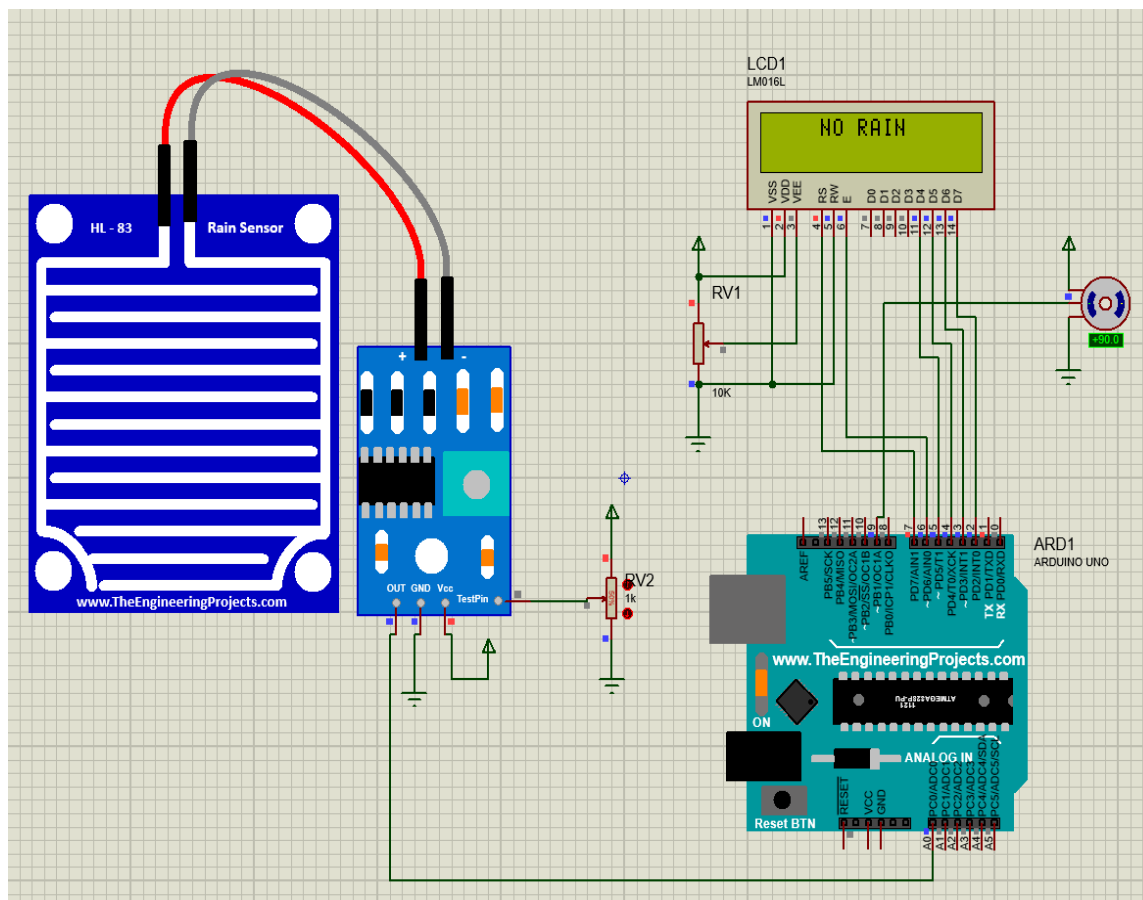


## 6. Code chương trình:

```
7.  #include <Servo.h>
8.  #include <Wire.h>
9.  #include <LiquidCrystal.h>
10.
11.  LiquidCrystal lcd(7,6,5,4,3,2);
12.  Servo myservo;
13.
14.  int pos = 0;
15.  int sensorValue = 0;
16.
17.  void setup() {
18.      Serial.begin(9600);
19.      lcd.begin(16,2);
20.      myservo.attach(9);
21.      lcd.clear();
22.      lcd.setCursor(0,0);
23.      lcd.print("  AUTOMATIC ");
24.      lcd.setCursor(0,1);
25.      lcd.print("  CAR WIPER ");
26.      delay(5000);
27.  }
28.
29.  void loop()
30.  {
31.      sensorValue = analogRead(A0);
32.      Serial.println(sensorValue);
33.      if(sensorValue>800){
34.          myservo.write(180);
35.          lcd.clear();
36.          lcd.setCursor(0,0);
37.          lcd.print("  NO RAIN ");
38.          delay(1000);
39.      }
40.      if(sensorValue<=800 && sensorValue>600){
41.          lcd.print("AMOUNT: LOW ");
42.          lcd.setCursor(0,0);
43.          lcd.print("IT IS RAINING");
44.          lcd.setCursor(0,1);
45.          for (pos = 180; pos >= 0; pos-=1) {
46.              myservo.write(pos);
47.              delay(3);
48.          }
```

```
49.     for (pos = 0; pos <= 180; pos+=1) {
50.         myservo.write(pos);
51.         delay(3);
52.     }
53.     delay(2000);
54. }
55.
56. if(sensorValue<=600 && sensorValue>460){
57.     lcd.setCursor(0,0);
58.     lcd.print("IT IS RAINING  ");
59.     lcd.setCursor(0,1);
60.     lcd.print("AMOUNT: MEDIUM ");
61.     for (pos = 180; pos >= 0; pos-=1) {
62.         myservo.write(pos);
63.         delay(3);
64.     }
65.     for (pos = 0; pos <= 180; pos+=1) {
66.         myservo.write(pos);
67.         delay(3);
68.     }
69.     delay(1000);
70. }
71.
72. if(sensorValue<460){
73.     lcd.setCursor(0,0);
74.     lcd.print("IT IS RAINING  ");
75.     lcd.setCursor(0,1);
76.     lcd.print("AMOUNT: HIGH   ");
77.     for (pos = 180; pos >= 0; pos-=1) {
78.         myservo.write(pos);
79.         delay(3);
80.     }
81.     for (pos = 0; pos <= 180; pos+=1) {
82.         myservo.write(pos);
83.         delay(3);
84.     }
85.     delay(100);
86. }
87.
88. }
```

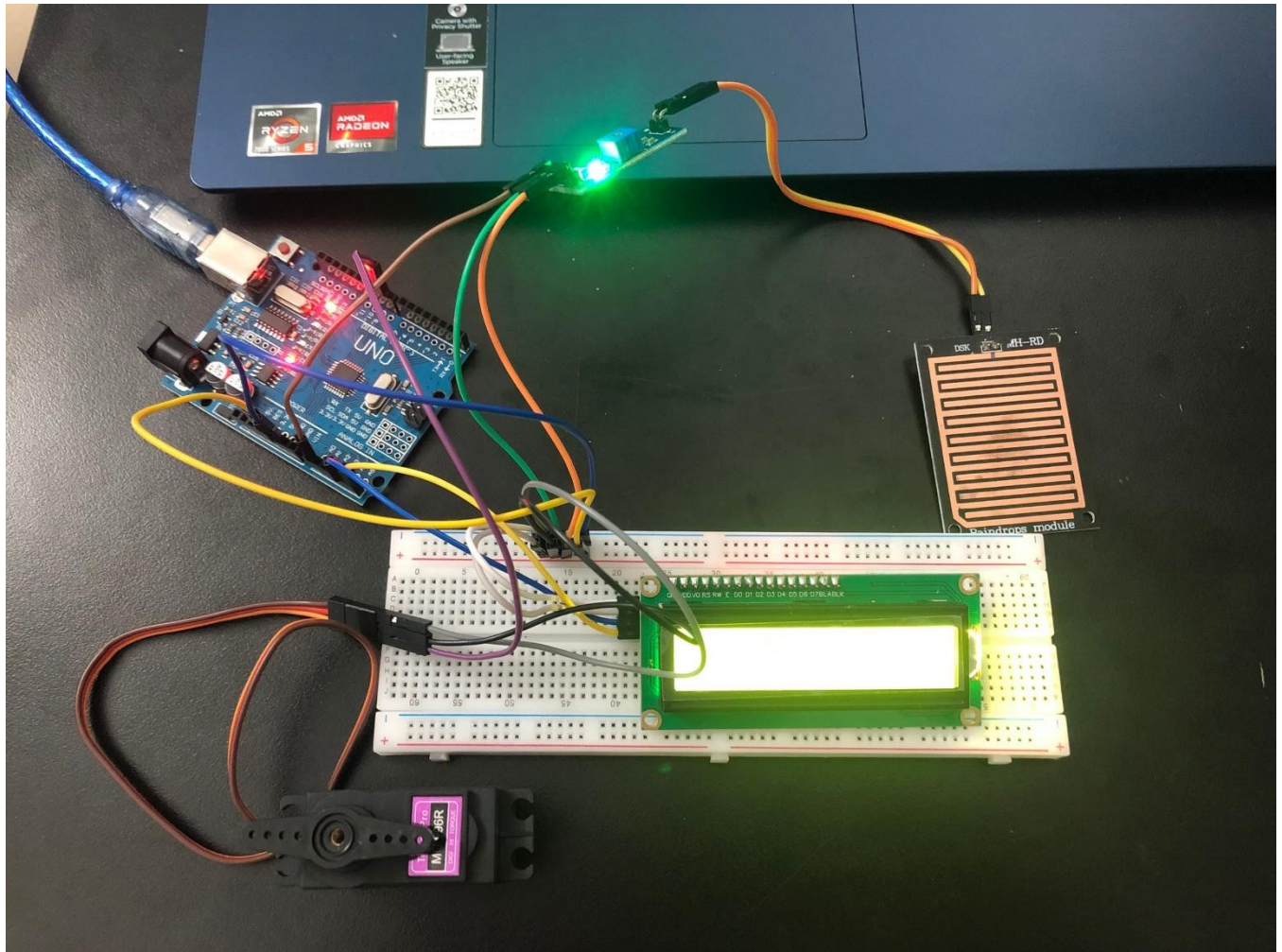
## 7. Mô phỏng:



Hình 3.6: Kết quả mô phỏng trên Protues

## 8. Mạch thực tế:

\* Thi công mạch:



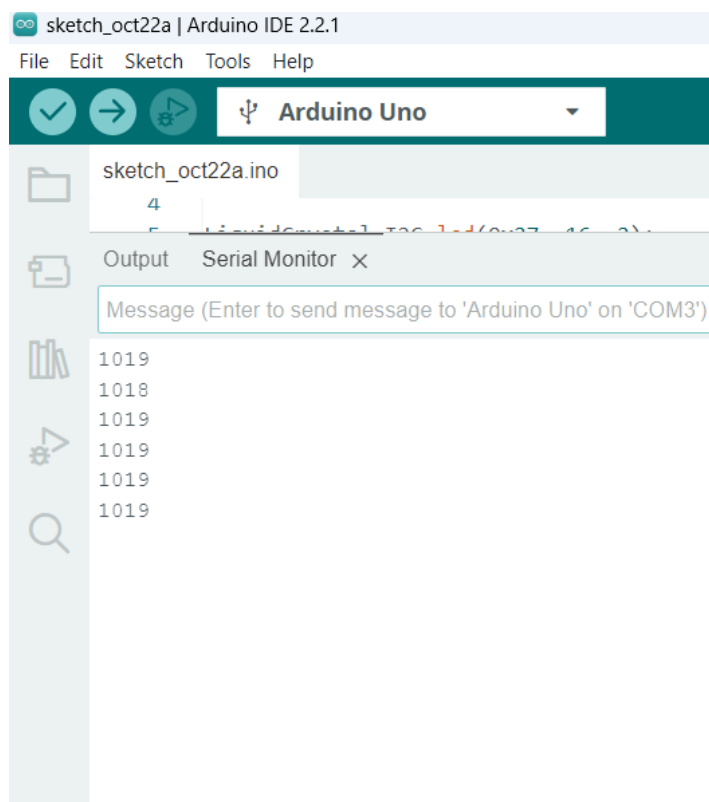


- \* Kết quả mô phỏng:
  - Giao diện màn hình lcd hiển thị khi khởi chạy mạch:

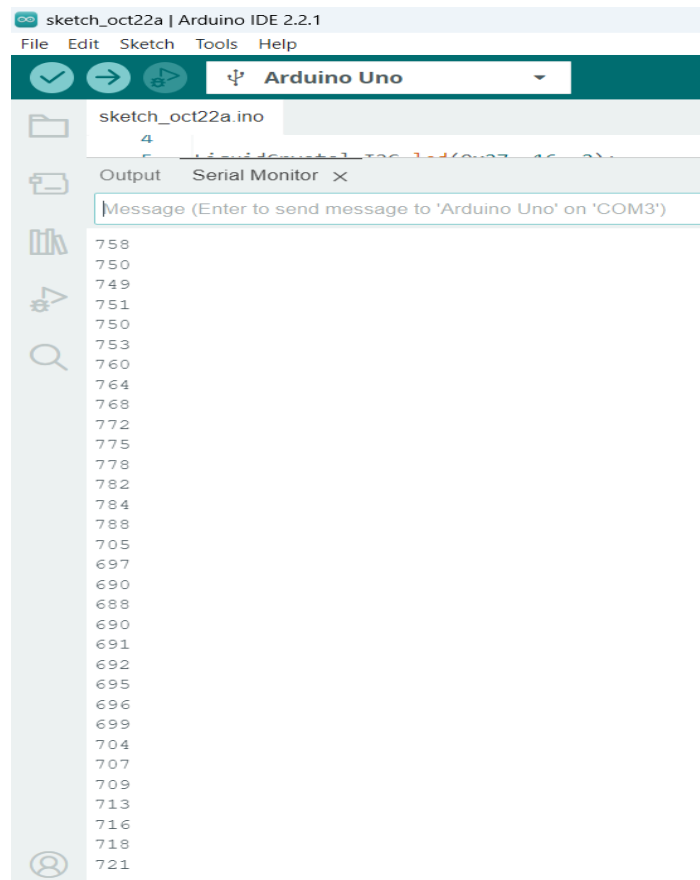




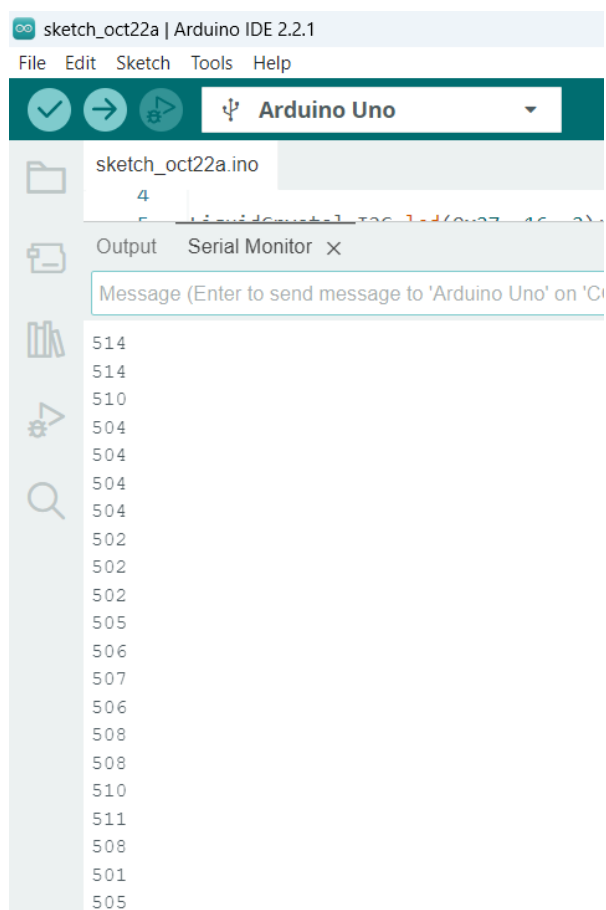
- Khi  $\text{sensorValue} > 800$  thì giao diện màn hình lcd hiển thị “NO RAIN”



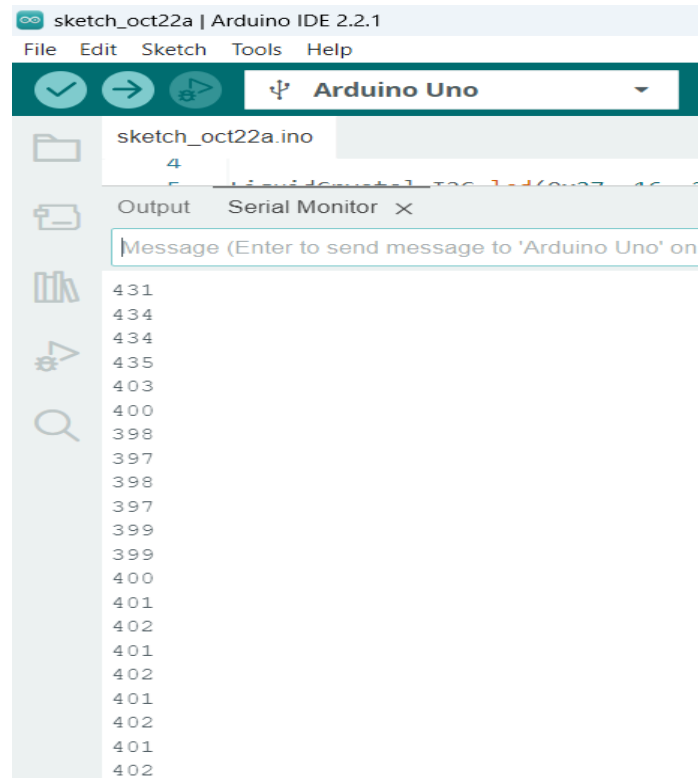
- Khi  $\text{sensorValue} \leq 800$  &  $\text{sensorValue} > 600$  thì giao diện màn hình lcd hiển thị “AMOUNT: LOW”



- Khi  $\text{sensorValue} \leq 600$  &  $\text{sensorValue} > 460$  thì giao diện màn hình lcd hiển thị “ AMOUNT: MEDIUM “



- Khi sensorValue < 460 thì giao diện màn hình lcd hiển thị “ AMOUNT: HIGH “



## Chương IV: KẾT LUẬN

Sau một thời gian tìm hiểu và nghiên cứu đề tài này chúng em đã thiết kế và thử nghiệm thành công hệ thống gạt nước ô tô tự động bằng cảm biến mưa dùng Arduino Uno.

### 1. Ưu/Nhược điểm:

#### ❖ Ưu điểm:

- Mạch gọn nhẹ, không cần mạch in PCB.
- Đáp ứng được những yêu cầu đặt ra.
- Chi phí thấp.
- Dễ dàng tìm mua linh kiện.

#### ❖ Nhược điểm:

- Chưa có giao diện trên máy tính và điện thoại để người chủ động điều khiển.
- Độ ổn định chưa cao.

### 2. Hướng phát triển:

- Khắc phục các nhược điểm đã được nêu ra.
- Áp dụng nguyên lý hoạt động vào những đề tài tương tự.
- Phát triển hệ thống tự lái trên ô tô.

Do thời gian và kiến thức còn hạn hẹp nên trong quá trình tìm hiểu, nghiên cứu và hoàn thiện đề tài không thể tránh khỏi những sai sót, rất mong nhận được sự đóng góp ý kiến từ quý thầy cô và các bạn để góp phần phát triển hoàn thiện đề tài này.

## TÀI LIỆU THAM KHẢO

- [1] P. J. Knaggs, "Sorting: A Distribution Theory".
- [2] J. Fraden, Handbook of Modern Sensors, New York, USA, 2010.
- [3] John David Warre, Harald Molle, "Arduino Robotics", New York, USA, 2011.