

TRƯỜNG ĐẠI HỌC ĐẠI NAM  
KHOA CÔNG NGHỆ THÔNG TIN



# BÀI TẬP LỚN

**TÊN MÔN HỌC: THÀNH PHỐ THÔNG MINH  
& NÔNG NGHIỆP THÔNG MINH**

**ĐỀ TÀI: GIÁM SÁT CÁC YẾU TỐ THỜI TIẾT ẢNH  
HƯỞNG ĐẾN NĂNG SUẤT CÂY TRỒNG**

**Giáo viên hướng dẫn: TS. Trần Đăng Công**

**Nguyễn Văn Nhân**

**Sinh viên thực hiện:**

Stt	Mã sv	Họ và tên	Lớp
1	1571020112	Bùi Khánh Hòa	CNTT 15-03

**Hà Nội, 2025**

TRƯỜNG ĐẠI HỌC ĐẠI NAM  
KHOA CÔNG NGHỆ THÔNG TIN



# BÀI TẬP LỚN

**TÊN MÔN HỌC: THÀNH PHỐ THÔNG MINH  
& NÔNG NGHIỆP THÔNG MINH**

**ĐỀ TÀI: GIÁM SÁT CÁC YẾU TỐ THỜI TIẾT ẢNH  
HƯỞNG ĐẾN NĂNG SUẤT CÂY TRỒNG**

**Giáo viên hướng dẫn: TS. Trần Đăng Công**

**Nguyễn Văn Nhân**

STT	Mã Sinh Viên	Họ và Tên	Ngày Sinh	Điểm	
				Bảng Số	Bảng Chữ
1	1571020112	Bùi Khánh Hòa	02/10/2003		

**CÁN BỘ CHẤM THI 1**

**CÁN BỘ CHẤM THI 2**

**Hà Nội, 2025**

## LỜI NÓI ĐẦU

Nông nghiệp là một trong những ngành kinh tế then chốt ở nhiều quốc gia, trong đó điều kiện thời tiết là yếu tố ảnh hưởng trực tiếp đến năng suất và chất lượng cây trồng. Việc theo dõi và phân tích các yếu tố thời tiết một cách chính xác và kịp thời là điều cần thiết để đưa ra các biện pháp canh tác phù hợp, hạn chế thiệt hại và nâng cao hiệu quả sản xuất.

Với sự phát triển mạnh mẽ của công nghệ cảm biến và trí tuệ nhân tạo, các hệ thống giám sát thông minh trong nông nghiệp ngày càng được ứng dụng rộng rãi. Đề tài "**Giám sát các yếu tố thời tiết ảnh hưởng đến năng suất cây trồng**" được thực hiện nhằm xây dựng một hệ thống thu thập dữ liệu thời tiết (nhiệt độ, độ ẩm không khí, độ ẩm đất, ánh sáng, lượng mưa) sử dụng các cảm biến kết hợp với vi điều khiển (UNO, ESP32). Dữ liệu sau khi thu thập sẽ được xử lý và phân tích bằng các thuật toán học máy để đánh giá và dự báo ảnh hưởng của thời tiết đến năng suất cây trồng, từ đó đưa ra các phương án điều chỉnh phù hợp.

Em xin gửi lời cảm ơn chân thành đến thầy Trần Đăng Công và thầy Nguyễn Văn Nhân đã tận tình hướng dẫn và hỗ trợ em trong suốt quá trình học tập và thực hiện. Em rất mong nhận được sự đánh giá và góp ý của quý thầy để đề tài được hoàn thiện hơn trong tương lai.

# MỤC LỤC

<b>LỜI NÓI ĐẦU .....</b>	<b>3</b>
<b>Chương 1: GIỚI THIỆU VỀ ĐỀ TÀI .....</b>	<b>1</b>
1.1. Lý do chọn đề tài.....	1
1.2. Mục tiêu nghiên cứu .....	1
1.3. Đối tượng và phạm vi đề tài.....	2
<b>Chương 2: CƠ SỞ LÝ THUYẾT .....</b>	<b>3</b>
2.1. Nông nghiệp thông minh (Smart Agriculture hay Smart Farming) .....	3
2.2. Các thành phần chính của nông nghiệp thông minh .....	3
2.3. Lợi ích của nông nghiệp thông minh .....	4
2.4. Kiến trúc nền tảng của đề tài.....	4
<b>Chương 3: Phần cứng và hệ thống sử dụng .....</b>	<b>6</b>
3.1. Vi điều khiển .....	6
3.2. Cảm biến sử dụng.....	6
3.3. Thư viện Python sử dụng trong hệ thống.....	8
<b>Chương 4: Kết nối phần cứng .....</b>	<b>10</b>
4.1. Sơ đồ hoạt động .....	10
4.2. Kết nối chi tiết các thiết bị với Arduino UNO và giao tiếp với ESP32 .....	10
4.3. Hình ảnh phần cứng kết nối thực tế .....	11
<b>Chương 5: Chương trình điều khiển phần cứng .....</b>	<b>12</b>
5.1. Chương trình thu thập dữ liệu cảm biến trên Arduino IDE.....	12
5.2. Chương trình đọc dữ liệu và dự đoán bằng Python trên Visal Studio Code..	15
<b>Chương 6: Kết quả và các bước hoạt động .....</b>	<b>19</b>
6.1. Quy trình tổng thể các bước hoạt động.....	19
6.2. Kết quả khởi chạy.....	20

<b>Chương 7: Đánh giá kết quả THỰC NGHIỆM.....</b>	<b>22</b>
7.1. Mục tiêu thực nghiệm .....	22
7.2. Thiết lập mô hình và dữ liệu.....	22
7.3. Kết quả thực nghiệm .....	23
7.4. Đánh giá độ chính xác .....	23
7.5. Kết luận .....	23
<b>Chương 8: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN .....</b>	<b>24</b>
8.1. Kết luận .....	24
8.2. Hạn chế .....	24
8.3. Hướng phát triển .....	24
<b>Tài liệu tham khảo .....</b>	<b>26</b>

# CHƯƠNG 1: GIỚI THIỆU VỀ ĐỀ TÀI

## 1.1. Lý do chọn đề tài

Trong bối cảnh biến đổi khí hậu ngày càng diễn biến phức tạp, thời tiết trở thành yếu tố khó kiểm soát và gây ảnh hưởng lớn đến năng suất cây trồng. Những thay đổi đột ngột về nhiệt độ, độ ẩm, ánh sáng hay lượng mưa có thể làm giảm sản lượng, ảnh hưởng đến chất lượng nông sản và gây thiệt hại nghiêm trọng cho người nông dân.

Trong khi đó, việc ứng dụng công nghệ trong nông nghiệp, đặc biệt là các hệ thống giám sát thông minh kết hợp với cảm biến và trí tuệ nhân tạo đã và đang chứng minh được hiệu quả cao trong việc theo dõi, phân tích và dự báo các điều kiện môi trường ảnh hưởng đến cây trồng.

Vì vậy, đề tài "**Giám sát các yếu tố thời tiết ảnh hưởng đến năng suất cây trồng**" mong muốn xây dựng một hệ thống giám sát thời tiết tự động, giúp thu thập và phân tích dữ liệu từ thực tế đồng ruộng. Hệ thống sẽ hỗ trợ người nông dân đưa ra quyết định kịp thời và chính xác trong quá trình canh tác, góp phần nâng cao năng suất và giảm thiểu rủi ro do thời tiết gây ra.

## 1.2. Mục tiêu nghiên cứu

Mục tiêu của đề tài là xây dựng một hệ thống giám sát tự động nhằm thu thập, phân tích và dự báo ảnh hưởng của các yếu tố thời tiết đến năng suất cây trồng. Cụ thể, em đặt ra các mục tiêu sau:

- Thiết kế và triển khai hệ thống thu thập dữ liệu môi trường bao gồm các thông số như: nhiệt độ, độ ẩm không khí, độ ẩm đất, ánh sáng, lưu lượng nước.
- Ứng dụng vi điều khiển (Arduino UNO, ESP32) và các cảm biến phù hợp để đo lường chính xác các yếu tố thời tiết.
- Xây dựng phần mềm thu thập, lưu trữ và xử lý dữ liệu từ các cảm biến.
- Áp dụng mô hình học máy để phân tích và dự báo tác động của điều kiện thời tiết đến năng suất cây trồng.

- Đề xuất các giải pháp hoặc cảnh báo kịp thời giúp người nông dân điều chỉnh phương pháp canh tác phù hợp với điều kiện môi trường.

### **1.3. Đối tượng và phạm vi đề tài**

#### **1). Đối tượng nghiên cứu:**

Đề tài tập trung nghiên cứu các yếu tố thời tiết ảnh hưởng trực tiếp đến năng suất cây trồng, bao gồm: nhiệt độ, độ ẩm không khí, độ ẩm đất, ánh sáng, lưu lượng nước. Bên cạnh đó, em cũng nghiên cứu việc ứng dụng vi điều khiển (Arduino UNO, ESP32), cảm biến môi trường và thuật toán học máy để thu thập, xử lý và phân tích dữ liệu từ thực tế.

#### **2). Phạm vi thực hiện:**

- Hệ thống được xây dựng và thử nghiệm trong phạm vi mô hình nhỏ tại quanh khuôn viên trường đại học Đại Nam.
- Dữ liệu thu thập được giới hạn trong khoảng thời gian ngắn và trong điều kiện môi trường thực tế tại Hà Nội.
- Mô hình học máy áp dụng ở mức cơ bản, chủ yếu để phân tích xu hướng và đưa ra dự báo sơ bộ về ảnh hưởng của thời tiết đến cây trồng.
- Đề tài không đi sâu vào phân tích theo từng loại cây cụ thể mà tập trung vào xây dựng một nền tảng hệ thống có thể mở rộng và tùy chỉnh sau này.

## CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

### 2.1. Nông nghiệp thông minh (Smart Agriculture hay Smart Farming)

Nông nghiệp thông minh (Smart Agriculture hay Smart Farming) là việc ứng dụng công nghệ hiện đại vào sản xuất nông nghiệp nhằm tăng năng suất, tối ưu hóa chi phí và giảm tác động tiêu cực đến môi trường. Đây là xu hướng tất yếu trong bối cảnh biến đổi khí hậu, thiếu hụt lao động và nhu cầu tiêu dùng ngày càng cao.

### 2.2. Các thành phần chính của nông nghiệp thông minh

Cảm biến và IoT (Internet of Things):

- Giám sát các chỉ số như nhiệt độ, độ ẩm không khí, ánh sáng, độ ẩm đất, pH, EC, v.v.
- Ví dụ: Cảm biến BH1750 (ánh sáng), DHT22 (nhiệt độ - độ ẩm), cảm biến độ ẩm đất, v.v.

Trí tuệ nhân tạo (AI) & học máy (ML):

- Dự đoán năng suất, sâu bệnh, lịch tưới tiêu hợp lý, v.v.
- Phân tích dữ liệu cảm biến để đưa ra quyết định tối ưu.

Hệ thống tự động:

- Tưới tiêu tự động, bón phân định lượng, điều khiển máy nông nghiệp không người lái.

Hệ thống giám sát từ xa:

- Giao diện web/app để theo dõi dữ liệu thời gian thực, nhận cảnh báo và điều khiển thiết bị từ xa.

Hệ thống thông tin địa lý (GIS) & vệ tinh:

- Theo dõi sức khỏe cây trồng, tình trạng đất, mô hình thời tiết trên diện rộng.



### 2.3. Lợi ích của nông nghiệp thông minh

Tăng năng suất	Tối ưu hóa việc tưới, bón phân, canh tác
Tiết kiệm tài nguyên	Sử dụng nước, phân bón, thuốc bảo vệ thực vật hiệu quả hơn
Quyết định chính xác	Nhờ vào dữ liệu thời gian thực và dự báo AI
Giảm chi phí	Tự động hóa giúp giảm công lao động và hao hụt
Thân thiện môi trường	Hạn chế ô nhiễm đất, nước và khí quyển

### 2.4. Kiến thức nền tảng của đề tài

#### 1). Ảnh hưởng của các yếu tố thời tiết đến cây trồng

- **Nhiệt độ:** Là yếu tố quyết định tốc độ sinh trưởng và phát triển của cây. Nhiệt độ quá cao hoặc quá thấp đều ảnh hưởng tiêu cực đến quá trình trao đổi chất, ra hoa và đậu quả.
- **Độ ẩm không khí:** Ảnh hưởng đến quá trình quang hợp, hô hấp và sự bay hơi nước của cây. Độ ẩm thấp làm cây mất nước nhanh, trong khi độ ẩm quá cao dễ phát sinh sâu bệnh.
- **Độ ẩm đất:** Là yếu tố rất quan trọng vì ảnh hưởng trực tiếp đến khả năng hút nước và chất dinh dưỡng của cây. Đất quá khô làm cây héo, đất quá ẩm làm úng rễ.
- **Ánh sáng:** Là nguồn năng lượng để cây thực hiện quang hợp. Cường độ và thời gian chiếu sáng phù hợp sẽ giúp cây phát triển tốt.
- **Lưu lượng nước:** Lượng nước tưới ảnh hưởng đến độ ẩm đất và khả năng sinh trưởng. Cần được điều chỉnh theo nhu cầu cây và điều kiện thời tiết thực tế.

#### 2). Vi điều khiển và cảm biến môi trường

Arduino UNO và ESP32: Là hai loại vi điều khiển phổ biến, dễ lập trình và phù hợp với các ứng dụng thu thập dữ liệu từ cảm biến. ESP32 hỗ trợ kết nối WiFi nên phù hợp để truyền dữ liệu về máy chủ hoặc lưu trữ đám mây.

Các loại cảm biến sử dụng:

- Cảm biến nhiệt độ, độ ẩm không khí (DHT11/DHT22): Đo nhiệt độ và độ ẩm môi trường.
- Cảm biến độ ẩm đất: Đo độ ẩm trong đất giúp xác định điều kiện nước tưới.
- Cảm biến ánh sáng BH1750: Đo cường độ ánh sáng môi trường.
- Cảm biến lưu lượng nước YF-S201: Đo lượng nước tưới thực tế.

### **3). Học máy trong phân tích dữ liệu nông nghiệp**

Học máy (machine learning) là phương pháp phân tích dữ liệu giúp hệ thống học từ dữ liệu thực tế để đưa ra dự đoán hoặc quyết định mà không cần lập trình rõ ràng từng bước.

Trong đề tài này, em sử dụng học máy để phân tích mối quan hệ giữa các yếu tố thời tiết (nhiệt độ, độ ẩm, ánh sáng, lưu lượng nước...) và năng suất cây trồng. Việc phân tích này giúp dự đoán xu hướng tăng trưởng, phát hiện bất thường và hỗ trợ người nông dân điều chỉnh kỹ thuật canh tác phù hợp.

## CHƯƠNG 3: PHẦN CỨNG VÀ HỆ THỐNG SỬ DỤNG

### 3.1. Vi điều khiển

Arduino UNO: Dùng để thu thập dữ liệu từ các cảm biến analog như cảm biến độ ẩm đất, cảm biến lưu lượng nước.

ESP32: Đóng vai trò chính trong việc xử lý và truyền dữ liệu. ESP32 có khả năng kết nối WiFi, giúp gửi dữ liệu lên máy tính hoặc lưu trữ đám mây.

### 3.2. Cảm biến sử dụng

#### 1). Cảm biến nhiệt độ và độ ẩm không khí (DHT11 hoặc DHT22)

- Thư viện sử dụng: DHT.h
- Chức năng: Giao tiếp và đọc dữ liệu nhiệt độ (°C) và độ ẩm (%) từ cảm biến.
- Cài đặt: Vào Arduino IDE → Library Manager → tìm và cài "DHT sensor library by Adafruit"

Code ví dụ:

```
#include <DHT.h>
```

```
#define DHTPIN 2      // Chân kết nối tín hiệu
```

```
#define DHTTYPE DHT22 // Hoặc DHT11 tùy cảm biến
```

```
DHT dht(DHTPIN, DHTTYPE);
```

#### 2). Cảm biến ánh sáng BH1750

- Thư viện sử dụng: BH1750.h và Wire.h
- Chức năng: Đọc cường độ ánh sáng môi trường theo đơn vị lux.
- Cài đặt: Library Manager → tìm và cài "BH1750 by Christopher Laws"

Code ví dụ:

```
#include <Wire.h>
```

```
#include <BH1750.h>
BH1750 lightMeter;
void setup() {
  Wire.begin();
  lightMeter.begin();
}
```

### 3). Cảm biến độ ẩm đất (analog)

- Thư viện: Không cần thư viện
- Chức năng: Đọc giá trị điện áp analog thể hiện độ ẩm của đất.
- Giao tiếp: Đọc trực tiếp bằng analogRead()

Code ví dụ:

```
int doAmDat = analogRead(A0); // A0 là chân kết nối cảm biến
```

### 4). Cảm biến lưu lượng nước (YF-S201 hoặc tương tự)

- Thư viện: Sseed\_Arduino\_FlowSensor hoặc có thể đo thủ công bằng attachInterrupt()
- Chức năng: Đếm xung trong một khoảng thời gian để tính lưu lượng nước.
- Ví dụ đo thủ công:

Code ví dụ:

```
volatile int pulseCount = 0;
void IRAM_ATTR countPulse() {
  pulseCount++;
}
void setup() {
  pinMode(2, INPUT_PULLUP); // Chân tín hiệu cảm biến
```

```
attachInterrupt(digitalPinToInterrupt(2), countPulse, RISING);  
}
```

### **3.3. Thư viện Python sử dụng trong hệ thống**

Để xử lý dữ liệu cảm biến, xây dựng giao diện giám sát và áp dụng mô hình học máy, em đã sử dụng các thư viện Python sau:

#### **1). Thư viện chuẩn (Built-in)**

- Tên thư viện: csv, time, sys, unicodedata
- Chức năng: Đọc/ghi file CSV, thao tác thời gian thực, xử lý chuỗi và mã hóa ký tự tiếng Việt.
- Cài đặt: Có sẵn trong Python, không cần cài

#### **2). Giao tiếp thiết bị**

- Tên thư viện: serial
- Chức năng: Giao tiếp với Arduino/ESP32 qua cổng COM để thu thập dữ liệu từ cảm biến.
- Cài đặt: pip install pyserial

#### **3). Giao diện người dùng (GUI)**

- Tên thư viện: tkinter
- Chức năng: Xây dựng giao diện đồ họa đơn giản để hiển thị và thao tác với dữ liệu.
- Cài đặt: Tích hợp sẵn trong Python

#### **4). Xử lý và phân tích dữ liệu**

- Tên thư viện: pandas, numpy
- Chức năng: Xử lý dữ liệu dạng bảng, thống kê, tính toán ma trận, xử lý dữ liệu huấn luyện mô hình.
- Cài đặt: pip install pandas numpy

## **5). Học máy (Machine Learning)**

- Tên thư viện: scikit-learn (LinearRegression)
- Chức năng: Xây dựng mô hình hồi quy tuyến tính để phân tích và dự đoán dựa trên dữ liệu cảm biến.
- Cài đặt: `pip install scikit-learn`

## CHƯƠNG 4: KẾT NỐI PHẦN CỨNG

### 4.1. Sơ đồ hoạt động

Hệ thống sử dụng Arduino UNO để thu thập tín hiệu từ các cảm biến, sau đó truyền dữ liệu sang ESP32 để xử lý và gửi về máy tính. Cách kết nối đảm bảo dễ triển khai và phù hợp với thiết bị có sẵn.

### 4.2. Kết nối chi tiết các thiết bị với Arduino UNO và giao tiếp với ESP32

#### 1). Thiết bị với Arduino UNO

Thiết bị	Chân kết nối với Arduino UNO	Ghi chú / Từ mã lệnh
Cảm biến nhiệt độ, độ ẩm (DHT11)	D4	<code>#define DHTPIN 4</code>
Cảm biến ánh sáng BH1750	SDA → A4, SCL → A5	Dùng I2C ( <code>Wire.begin()</code> , <code>lightMeter.begin()</code> )
Cảm biến độ ẩm đất (analog)	A0	<code>#define SOIL_PIN A0</code> , đọc bằng <code>analogRead(SOIL_PIN)</code>
Rơ-le điều khiển bơm	D7	<code>#define RELAY_PIN 7</code> , <code>digitalWrite(RELAY_PIN, ...)</code>
Cảm biến lưu lượng nước YF-S201	D2 (INT0)	<code>#define FLOW_SENSOR 2</code> , dùng <code>attachInterrupt(..., flowISR, ...)</code>

#### 2). Giao tiếp với ESP32

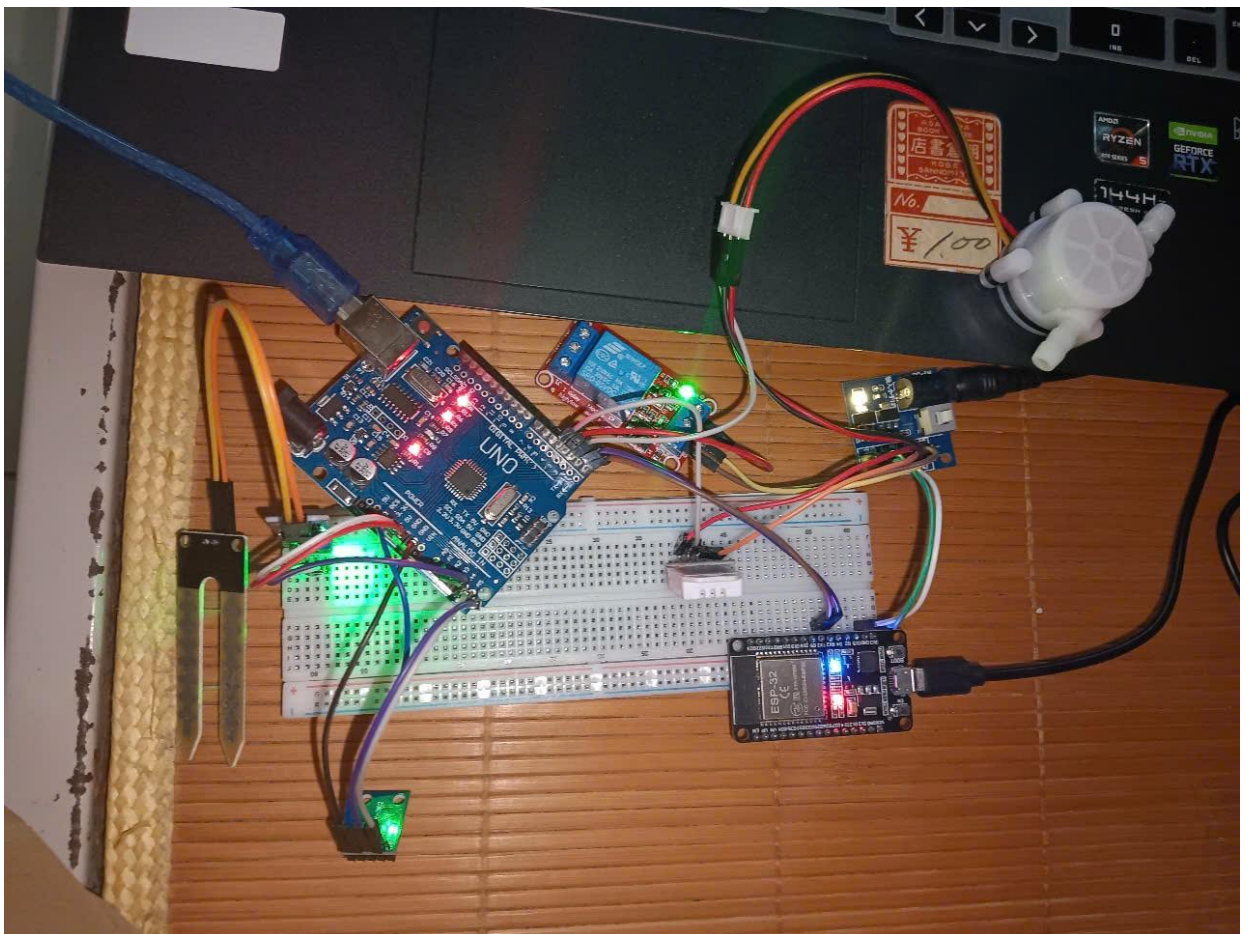
- UART (Serial):
  - Arduino UNO gửi dữ liệu qua Serial tốc độ 9600.
  - ESP32 cần kết nối:
    - TX (UNO) → RX (ESP32)
    - GND (UNO) ↔ GND (ESP32)

*(UNO không nhận dữ liệu nên không cần nối RX của UNO trong trường hợp chỉ truyền 1 chiều)*

### 3). Nguồn cấp điện

- Arduino UNO: Kết nối qua cổng USB máy tính hoặc dùng nguồn 5V từ ESP32.
- ESP32: Cấp nguồn qua cổng USB hoặc Vin (3.3V–5V).
- Các cảm biến sử dụng nguồn 3.3V hoặc 5V tùy loại (phù hợp với cả UNO và ESP32).

### 4.3. Hình ảnh phần cứng kết nối thực tế

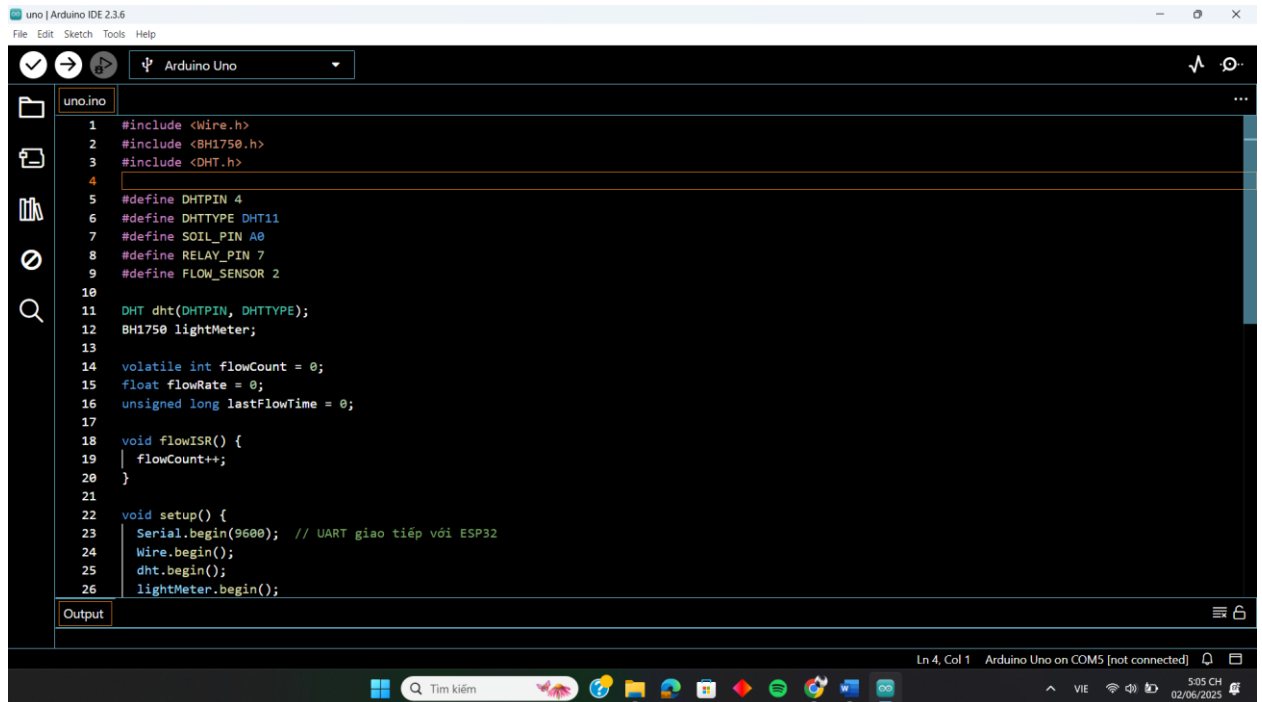




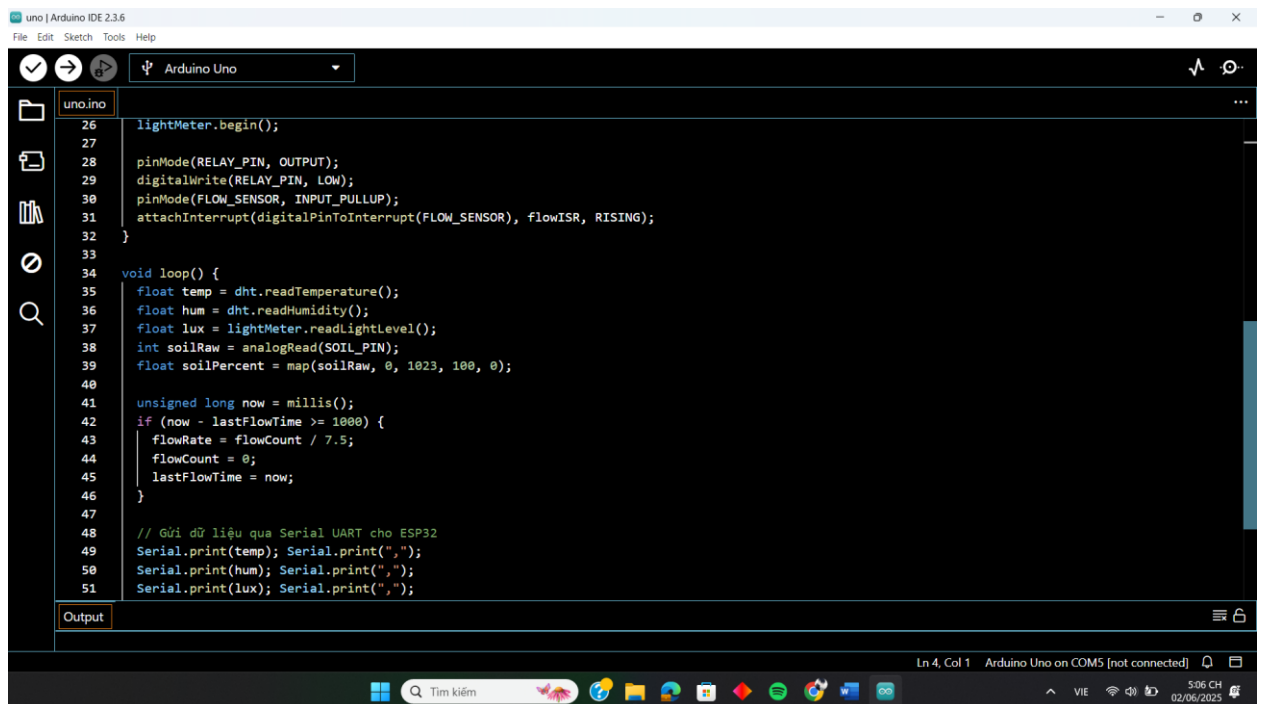
# CHƯƠNG 5: CHƯƠNG TRÌNH ĐIỀU KHIỂN PHẦN CỨNG

## 5.1. Chương trình thu thập dữ liệu cảm biến trên Arduino IDE

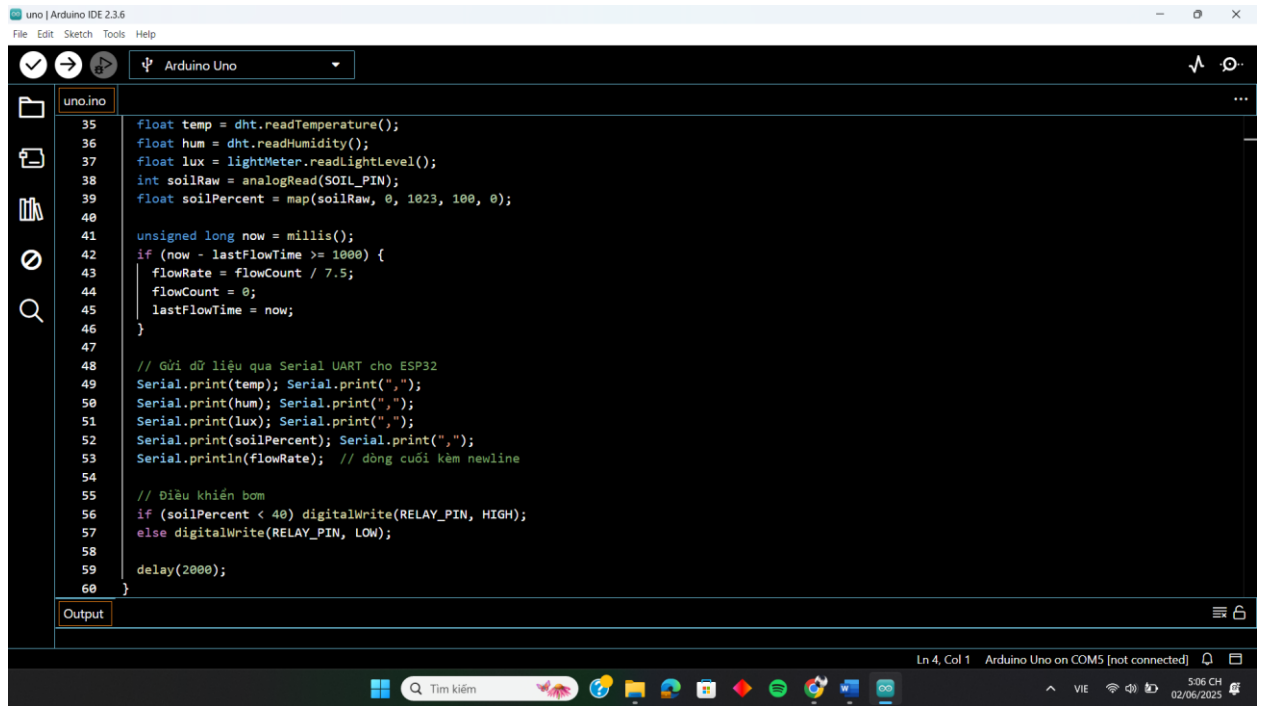
### 1). Code nạp vào Arduino Uno



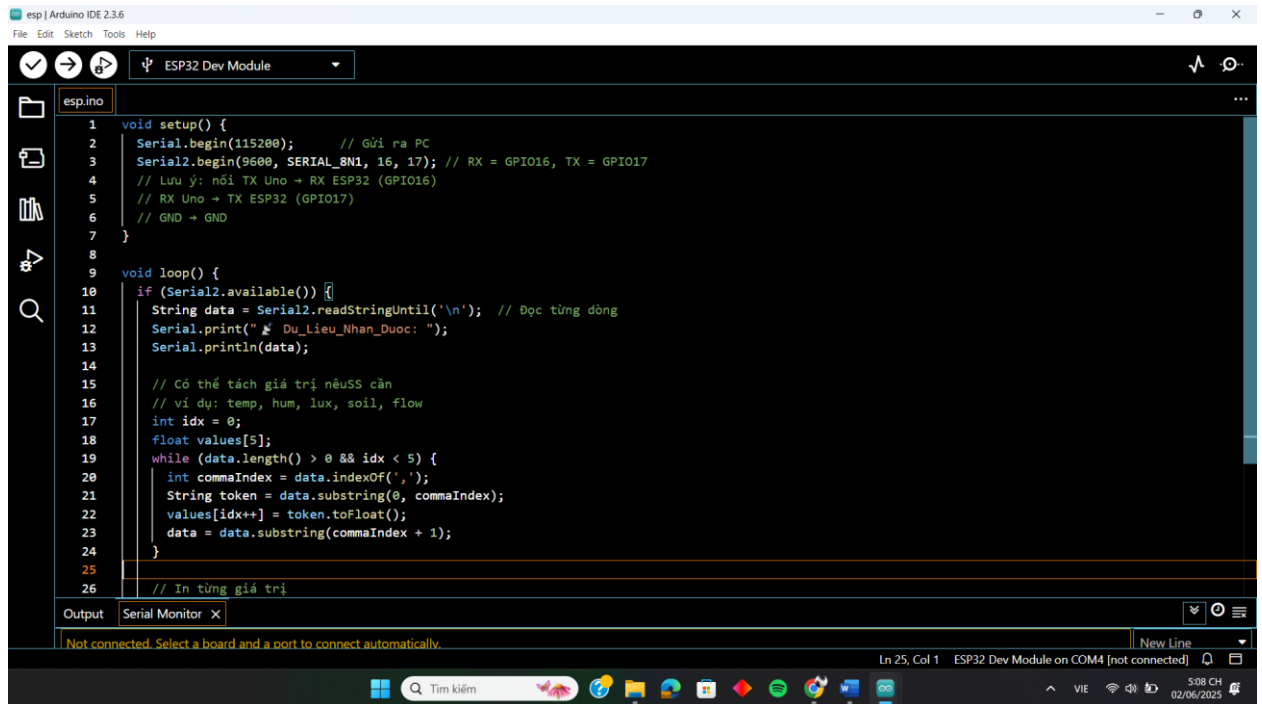
```
1 #include <Wire.h>
2 #include <BH1750.h>
3 #include <DHT.h>
4
5 #define DHTPIN 4
6 #define DHTTYPE DHT11
7 #define SOIL_PIN A0
8 #define RELAY_PIN 7
9 #define FLOW_SENSOR 2
10
11 DHT dht(DHTPIN, DHTTYPE);
12 BH1750 lightMeter;
13
14 volatile int flowCount = 0;
15 float flowRate = 0;
16 unsigned long lastFlowTime = 0;
17
18 void flowISR() {
19   flowCount++;
20 }
21
22 void setup() {
23   Serial.begin(9600); // UART giao tiếp với ESP32
24   Wire.begin();
25   dht.begin();
26   lightMeter.begin();
```



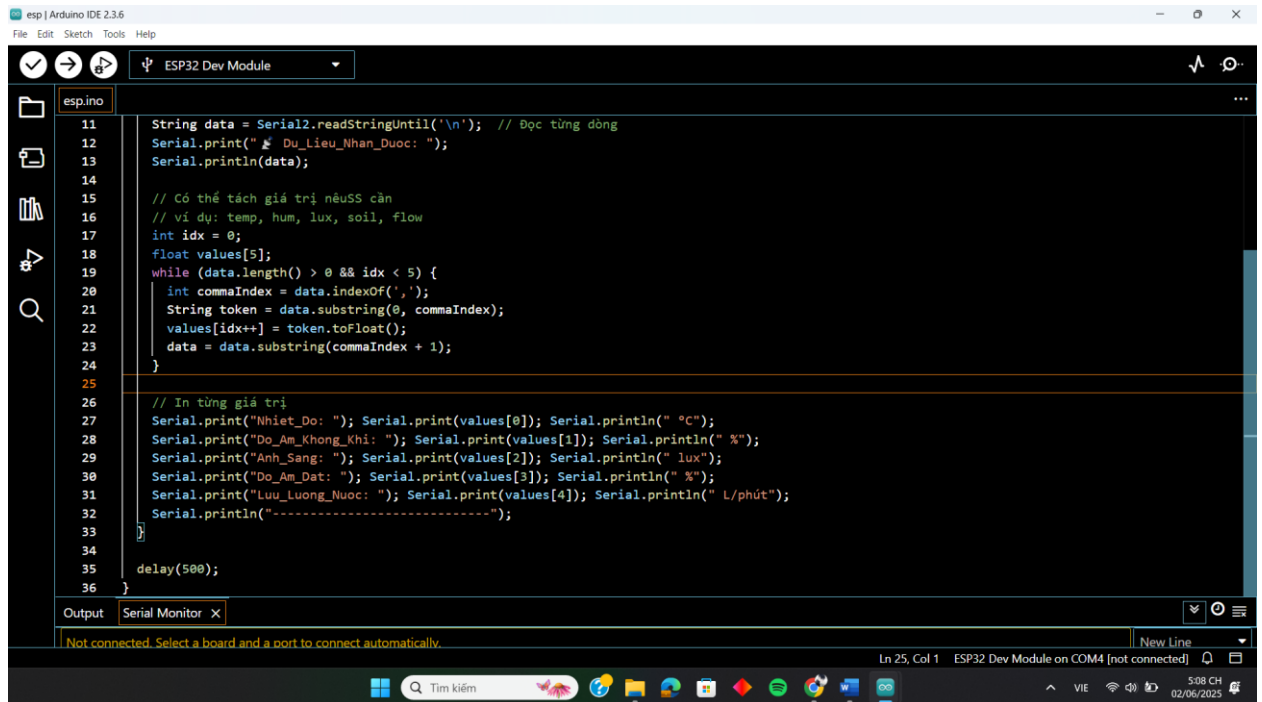
```
26   lightMeter.begin();
27
28   pinMode(RELAY_PIN, OUTPUT);
29   digitalWrite(RELAY_PIN, LOW);
30   pinMode(FLOW_SENSOR, INPUT_PULLUP);
31   attachInterrupt(digitalPinToInterrupt(FLOW_SENSOR), flowISR, RISING);
32 }
33
34 void loop() {
35   float temp = dht.readTemperature();
36   float hum = dht.readHumidity();
37   float lux = lightMeter.readLightLevel();
38   int soilRaw = analogRead(SOIL_PIN);
39   float soilPercent = map(soilRaw, 0, 1023, 100, 0);
40
41   unsigned long now = millis();
42   if (now - lastFlowTime >= 1000) {
43     flowRate = flowCount / 7.5;
44     flowCount = 0;
45     lastFlowTime = now;
46   }
47
48   // Gửi dữ liệu qua Serial UART cho ESP32
49   Serial.print(temp); Serial.print(",");
50   Serial.print(hum); Serial.print(",");
51   Serial.print(lux); Serial.print(",");
```



## 2). Code nạp vào Esp32



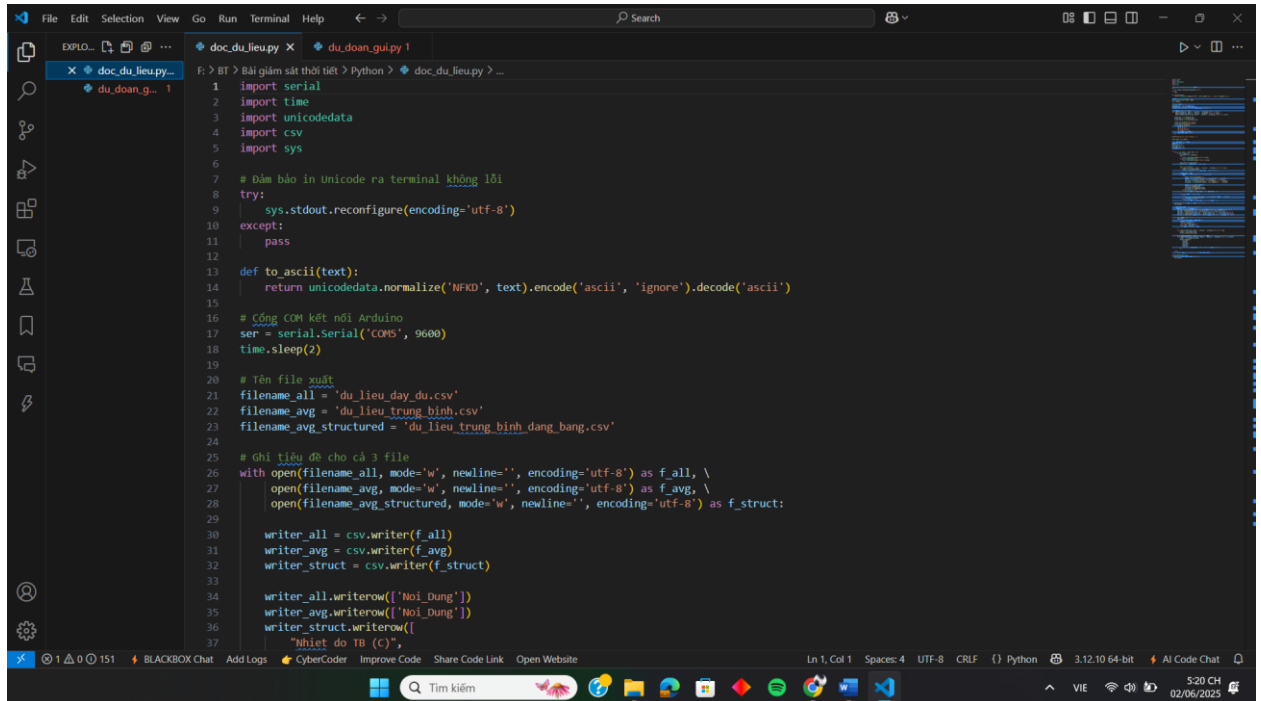
```
1 void setup() {
2   Serial.begin(115200); // Gửi ra PC
3   Serial2.begin(9600, SERIAL_8N1, 16, 17); // RX = GPIO16, TX = GPIO17
4   // Lưu ý: nối TX Uno → RX ESP32 (GPIO16)
5   // RX Uno → TX ESP32 (GPIO17)
6   // GND → GND
7 }
8
9 void loop() {
10  if (Serial2.available()) {
11    String data = Serial2.readStringUntil('\n'); // Đọc từng dòng
12    Serial.print("Đu_Lieu_Nhan_Duoc: ");
13    Serial.println(data);
14
15    // Có thể tách giá trị nếu cần
16    // ví dụ: temp, hum, lux, soil, flow
17    int idx = 0;
18    float values[5];
19    while (data.length() > 0 && idx < 5) {
20      int commaIndex = data.indexOf(',');
21      String token = data.substring(0, commaIndex);
22      values[idx++] = token.toFloat();
23      data = data.substring(commaIndex + 1);
24    }
25
26    // In từng giá trị
```



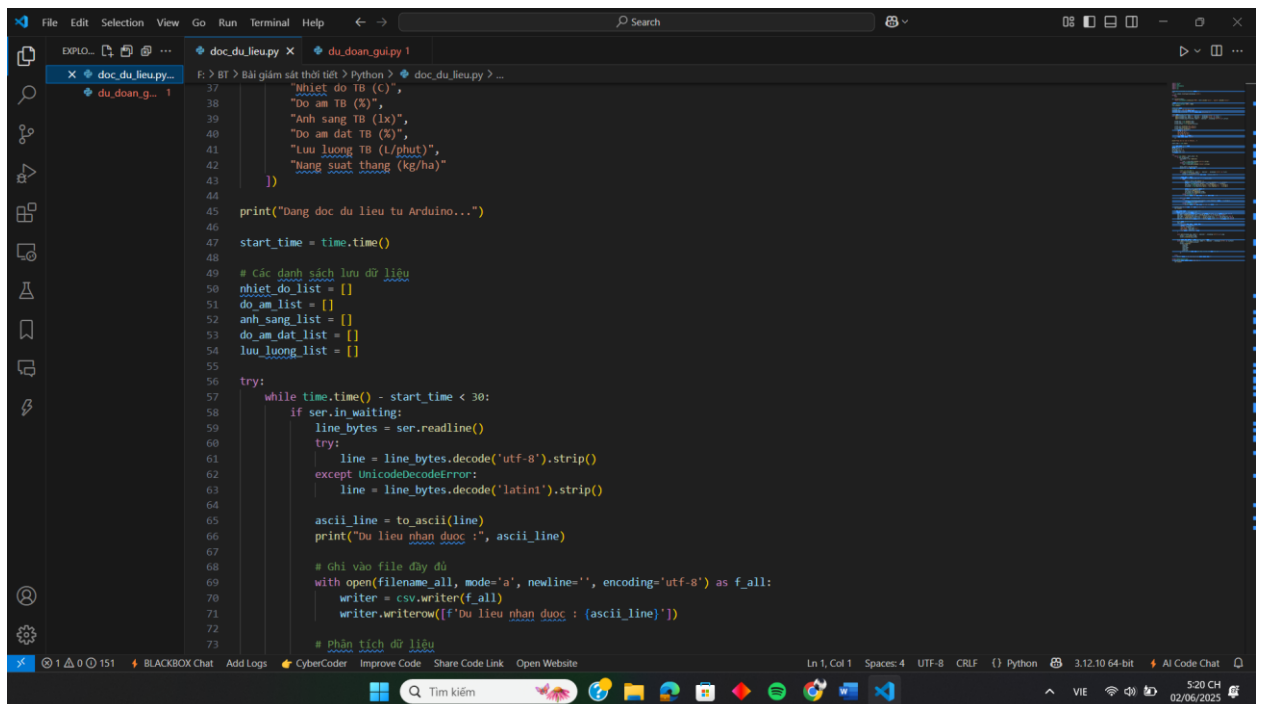
```
11 String data = Serial2.readStringUntil('\n'); // Đọc từng dòng
12 Serial.print("Đu_Lieu_Nhan_Duoc: ");
13 Serial.println(data);
14
15 // Có thể tách giá trị nếu cần
16 // ví dụ: temp, hum, lux, soil, flow
17 int idx = 0;
18 float values[5];
19 while (data.length() > 0 && idx < 5) {
20   int commaIndex = data.indexOf(',');
21   String token = data.substring(0, commaIndex);
22   values[idx++] = token.toFloat();
23   data = data.substring(commaIndex + 1);
24 }
25
26 // In từng giá trị
27 Serial.print("Nhiet_Do: "); Serial.print(values[0]); Serial.println(" °C");
28 Serial.print("Do_Am_Khong_Khi: "); Serial.print(values[1]); Serial.println(" %");
29 Serial.print("Anh_Sang: "); Serial.print(values[2]); Serial.println(" lux");
30 Serial.print("Do_Am_Dat: "); Serial.print(values[3]); Serial.println(" %");
31 Serial.print("Luoi_Luong_Nuoc: "); Serial.print(values[4]); Serial.println(" L/phút");
32 Serial.println("-----");
33
34
35 delay(500);
36 }
```

## 5.2. Chương trình đọc dữ liệu và dự đoán bằng Python trên Visal Studio Code

### 1). Đọc dữ liệu từ thiết bị và xuất file CSV



```
1 import serial
2 import time
3 import unicodedata
4 import csv
5 import sys
6
7 # Đảm bảo in Unicode ra terminal không lỗi
8 try:
9     sys.stdout.reconfigure(encoding='utf-8')
10 except:
11     pass
12
13 def to_ascii(text):
14     return unicodedata.normalize('NFKD', text).encode('ascii', 'ignore').decode('ascii')
15
16 # Cổng COM kết nối Arduino
17 ser = serial.Serial('COM5', 9600)
18 time.sleep(2)
19
20 # Tên file xuất
21 filename_all = 'du_lieu_day_du.csv'
22 filename_avg = 'du_lieu_trung_binh.csv'
23 filename_avg_structured = 'du_lieu_trung_binh_dang_bang.csv'
24
25 # Ghi tiêu đề cho cả 3 file
26 with open(filename_all, mode='w', newline='', encoding='utf-8') as f_all, \
27     open(filename_avg, mode='w', newline='', encoding='utf-8') as f_avg, \
28     open(filename_avg_structured, mode='w', newline='', encoding='utf-8') as f_struct:
29
30     writer_all = csv.writer(f_all)
31     writer_avg = csv.writer(f_avg)
32     writer_struct = csv.writer(f_struct)
33
34     writer_all.writerow(['Nội Dung'])
35     writer_avg.writerow(['Nội Dung'])
36     writer_struct.writerow([
37         "Nhiệt độ TB (C)",
38         "Độ ẩm TB (%)",
39         "Ánh sáng TB (lx)",
40         "Độ ẩm đất TB (%)",
41         "Lưu lượng TB (L/phút)",
42         "Năng suất tháng (kg/ha)"
43     ])
44
45 print("Đang đọc dữ liệu từ Arduino...")
46
47 start_time = time.time()
48
49 # Các danh sách lưu dữ liệu
50 nhiet_do_list = []
51 do_am_list = []
52 anh_sang_list = []
53 do_am_dat_list = []
54 luu_luong_list = []
55
56 try:
57     while time.time() - start_time < 30:
58         if ser.in_waiting:
59             line_bytes = ser.readline()
60             try:
61                 line = line_bytes.decode('utf-8').strip()
62             except UnicodeDecodeError:
63                 line = line_bytes.decode('latin1').strip()
64
65             ascii_line = to_ascii(line)
66             print("Dữ liệu nhận được :", ascii_line)
67
68             # Ghi vào file đầy đủ
69             with open(filename_all, mode='a', newline='', encoding='utf-8') as f_all:
70                 writer = csv.writer(f_all)
71                 writer.writerow([f'Dữ liệu nhận được : {ascii_line}'])
72
73             # Phân tích dữ liệu
```



```
37         "Nhiệt độ TB (C)",
38         "Độ ẩm TB (%)",
39         "Ánh sáng TB (lx)",
40         "Độ ẩm đất TB (%)",
41         "Lưu lượng TB (L/phút)",
42         "Năng suất tháng (kg/ha)"
43     ])
44
45 print("Đang đọc dữ liệu từ Arduino...")
46
47 start_time = time.time()
48
49 # Các danh sách lưu dữ liệu
50 nhiet_do_list = []
51 do_am_list = []
52 anh_sang_list = []
53 do_am_dat_list = []
54 luu_luong_list = []
55
56 try:
57     while time.time() - start_time < 30:
58         if ser.in_waiting:
59             line_bytes = ser.readline()
60             try:
61                 line = line_bytes.decode('utf-8').strip()
62             except UnicodeDecodeError:
63                 line = line_bytes.decode('latin1').strip()
64
65             ascii_line = to_ascii(line)
66             print("Dữ liệu nhận được :", ascii_line)
67
68             # Ghi vào file đầy đủ
69             with open(filename_all, mode='a', newline='', encoding='utf-8') as f_all:
70                 writer = csv.writer(f_all)
71                 writer.writerow([f'Dữ liệu nhận được : {ascii_line}'])
72
73             # Phân tích dữ liệu
```

```
File Edit Selection View Go Run Terminal Help
EXPLO... doc_du_lieu.py du_doan_guipy 1
X doc_du_lieu.py... F: > BT > Bài giám sát thời tiết > Python > doc_du_lieu.py > ...
du_doan_guipy 1
73
74 # phân tích dữ liệu
75 if "Nhiệt độ" in ascii_line and "L" in ascii_line:
76     try:
77         parts = ascii_line.split(',')
78         nhiet_do = float(parts[0].split(':')[1].replace('C', '').strip())
79         do_am = float(parts[1].split(':')[1].replace('X', '').strip())
80         anh_sang = float(parts[2].split(':')[1].replace('lx', '').strip())
81         do_am_dat = float(parts[3].split(':')[1].replace('X', '').strip())
82
83         nhiet_do_list.append(nhiet_do)
84         do_am_list.append(do_am)
85         anh_sang_list.append(anh_sang)
86         do_am_dat_list.append(do_am_dat)
87     except Exception as e:
88         print("Không thể phân tích dữ liệu thời tiết:", e)
89
90 elif "Lưu lượng" in ascii_line:
91     try:
92         value = float(ascii_line.split(':')[1].replace('L/phut', '').strip())
93         luu_luong_list.append(value)
94     except Exception as e:
95         print("Không thể phân tích dữ liệu lưu lượng:", e)
96
97 print("\nĐã hết 30 giây. Kết thúc đọc dữ liệu.")
98 ser.close()
99
100 # Tính trung bình và ghi vào cả hai file
101 if nhiet_do_list or luu_luong_list:
102     avg_temp = round(sum(nhiet_do_list) / len(nhiet_do_list), 2) if nhiet_do_list else 0
103     avg_hum = round(sum(do_am_list) / len(do_am_list), 2) if do_am_list else 0
104     avg_light = round(sum(anh_sang_list) / len(anh_sang_list), 2) if anh_sang_list else 0
105     avg_soil = round(sum(do_am_dat_list) / len(do_am_dat_list), 2) if do_am_dat_list else 0
106     avg_flow = round(sum(luu_luong_list) / len(luu_luong_list), 2) if luu_luong_list else 0
107
108 # Ghi dạng văn bản mô tả
109 avg_line = (
110     f'Dữ liệu nhận được : Trung bình -> '
111     f'Nhiệt độ: {avg_temp} C | '
112     f'Do ẩm: {avg_hum} % | '
113     f'Ánh sáng: {avg_light} lx | '
114     f'Do ẩm đất: {avg_soil} % | '
115     f'Lưu lượng: {avg_flow} L/phut'
116 )
117
118 with open(filename_avg, mode='a', newline='', encoding='utf-8') as f_avg:
119     writer = csv.writer(f_avg)
120     writer.writerow([avg_line])
121
122 # Ghi dạng bảng chuẩn để phân mềm học máy sử dụng
123 with open(filename_avg_structured, mode='a', newline='', encoding='utf-8') as f_struct:
124     writer = csv.writer(f_struct)
125     writer.writerow([
126         avg_temp,
127         avg_hum,
128         avg_light,
129         avg_soil,
130         avg_flow,
131         0 # Giá trị năng suất giả lập, có thể sửa bằng tay sau
132     ])
133
134 else:
135     print("Không có dữ liệu hợp lệ để tính trung bình.")
136
137 except KeyboardInterrupt:
138     print("Đã ngắt chương trình bằng tay.")
139     ser.close()
```

```
File Edit Selection View Go Run Terminal Help
EXPLO... doc_du_lieu.py du_doan_guipy 1
X doc_du_lieu.py... F: > BT > Bài giám sát thời tiết > Python > doc_du_lieu.py > ...
du_doan_guipy 1
103 avg_light = round(sum(anh_sang_list) / len(anh_sang_list), 2) if anh_sang_list else 0
104 avg_soil = round(sum(do_am_dat_list) / len(do_am_dat_list), 2) if do_am_dat_list else 0
105 avg_flow = round(sum(luu_luong_list) / len(luu_luong_list), 2) if luu_luong_list else 0
106
107 # Ghi dạng văn bản mô tả
108 avg_line = (
109     f'Dữ liệu nhận được : Trung bình -> '
110     f'Nhiệt độ: {avg_temp} C | '
111     f'Do ẩm: {avg_hum} % | '
112     f'Ánh sáng: {avg_light} lx | '
113     f'Do ẩm đất: {avg_soil} % | '
114     f'Lưu lượng: {avg_flow} L/phut'
115 )
116
117 with open(filename_avg, mode='a', newline='', encoding='utf-8') as f_avg:
118     writer = csv.writer(f_avg)
119     writer.writerow([avg_line])
120
121 # Ghi dạng bảng chuẩn để phân mềm học máy sử dụng
122 with open(filename_avg_structured, mode='a', newline='', encoding='utf-8') as f_struct:
123     writer = csv.writer(f_struct)
124     writer.writerow([
125         avg_temp,
126         avg_hum,
127         avg_light,
128         avg_soil,
129         avg_flow,
130         0 # Giá trị năng suất giả lập, có thể sửa bằng tay sau
131     ])
132
133 else:
134     print("Không có dữ liệu hợp lệ để tính trung bình.")
135
136 except KeyboardInterrupt:
137     print("Đã ngắt chương trình bằng tay.")
138     ser.close()
139
```

## 2). Code dự đoán và so sánh với các năm

```
File Edit Selection View Go Run Terminal Help
EXPLO... du_doan_lieu.py du_doan_gui.py 1
F: > BT > Bài giám sát thời tiết > Python > du_doan_gui.py > build_and_predict
1 import tkinter as tk
2 from tkinter import filedialog, messagebox
3 import pandas as pd
4 from sklearn.linear_model import LinearRegression
5
6 # Hàm xử lý dự đoán năng suất
7
8 def build_and_predict(sim_file_path, real_file_path):
9     try:
10         # Đọc dữ liệu mô phỏng và thực tế
11         df_sim = pd.read_csv(sim_file_path)
12         df_real = pd.read_csv(real_file_path)
13
14         # Dữ liệu đầu vào thực tế
15         real_temp = df_real.iloc[0]["Nhiệt độ TB (C)"]
16         real_hum = df_real.iloc[0]["Độ ẩm TB (%)"]
17         real_light = df_real.iloc[0]["Ánh sáng TB (lx)"]
18         real_soil = df_real.iloc[0]["Độ ẩm đất TB (%)"]
19         real_flow = df_real.iloc[0]["Lưu lượng TB (l/phút)"]
20
21         real_input = [[real_temp, real_hum, real_light, real_soil, real_flow]]
22
23         # Dự đoán và so sánh theo từng năm
24         result_lines = []
25
26         for year in sorted(df_sim['Nam'].unique()):
27             df_year = df_sim[df_sim['Nam'] == year]
28             x = df_year[["Nhiệt độ TB (C)",
29                         "Độ ẩm TB (%)",
30                         "Ánh sáng TB (lx)",
31                         "Độ ẩm đất TB (%)",
32                         "Lưu lượng TB (l/phút)"]]
33             y = df_year["Năng suất tháng (kg/ha)"]
34
35             model = LinearRegression()
36
37             model.fit(x, y)
```

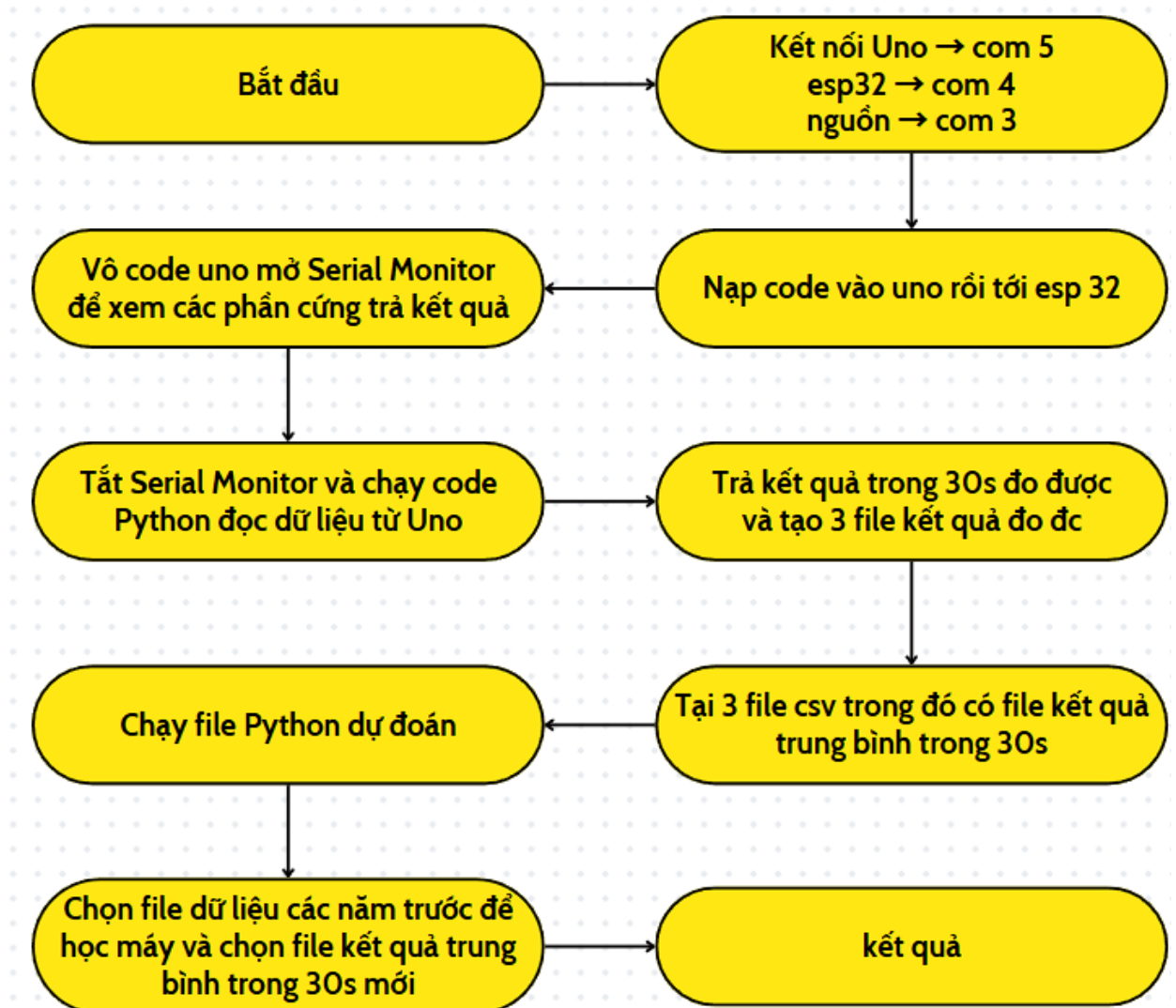
```
38
39
40         predicted_y = model.predict(real_input)[0]
41         avg_sim_y = y.mean()
42         delta_percent = ((predicted_y - avg_sim_y) / avg_sim_y) * 100
43
44         result_lines.append(f"📅 Năm (year):")
45         result_lines.append(f"📊 Năng suất dự đoán: (round(predicted_y, 2))")
46         result_lines.append(f"📊 Trung bình năm (year): (round(avg_sim_y, 2))")
47         result_lines.append(f"📊 % Chênh lệch: (round(delta_percent, 2))%")
48
49         # Lời khuyên
50         avg_year_temp = df_year["Nhiệt độ TB (C)"].mean()
51         if real_temp > avg_year_temp + 1:
52             result_lines.append(f"🔴 Cảnh báo: Năm nay nhiệt độ cao hơn trung bình rõ rệt. Cần chú ý tưới nước và che phủ.")
53         elif real_temp < avg_year_temp - 1:
54             result_lines.append(f"🔵 Lưu ý: Năm nay mát hơn trung bình. Có thể điều chỉnh lịch bón phân để tối ưu.")
55         else:
56             result_lines.append(f"🟢 Thời tiết năm nay tương đối ổn định so với trung bình năm.")
57
58         return result_lines
59
60     except Exception as e:
61         return [f"Lỗi: {str(e)}"]
62
63 # Giao diện tkinter
64
65 def run_gui():
66     def select_sim_file():
67         path = filedialog.askopenfilename(filetypes=[("CSV Files", "*.csv")])
68         if path:
69             sim_file_var.set(path)
70
71     def select_real_file():
72         path = filedialog.askopenfilename(filetypes=[("CSV Files", "*.csv")])
73         if path:
```

```
File Edit Selection View Go Run Terminal Help
doc_du_lieu.py du_doan_gui.py 1
F: > BT > Bài giảng sát thời tiết > Python > du_doan_gui.py > build_and_predict
65 def run_gui():
66     def select_real_file():
67         real_file_var.set(path)
68     def predict():
69         sim_path = sim_file_var.get()
70         real_path = real_file_var.get()
71         if not sim_path or not real_path:
72             messagebox.showwarning("Thiếu file", "Vui lòng chọn cả hai file CSV.")
73             return
74         result = build_and_predict(sim_path, real_path)
75         result_text.delete("1.0", tk.END)
76         for line in result:
77             result_text.insert(tk.END, line + "\n")
78     # Cửa sổ chính
79     root = tk.Tk()
80     root.title("Dự đoán năng suất cây trồng")
81     root.geometry("650x600")
82     sim_file_var = tk.StringVar()
83     real_file_var = tk.StringVar()
84     tk.Label(root, text="Chọn file mô phỏng (.csv):").pack()
85     tk.Entry(root, textvariable=sim_file_var, width=80).pack()
86     tk.Button(root, text="Duyệt...", command=select_sim_file).pack(pady=5)
87     tk.Label(root, text="Chọn file dữ liệu thực tế (.csv):").pack()
88     tk.Entry(root, textvariable=real_file_var, width=80).pack()
89     tk.Button(root, text="Duyệt...", command=select_real_file).pack(pady=5)
90     tk.Button(root, text="Dự đoán", command=predict, bg='green', fg='white').pack(pady=10)
91     result_text = tk.Text(root, height=30)
92     result_text.pack(fill=tk.BOTH, expand=True, padx=10, pady=10)
```

```
File Edit Selection View Go Run Terminal Help
doc_du_lieu.py du_doan_gui.py 1
F: > BT > Bài giảng sát thời tiết > Python > du_doan_gui.py > build_and_predict
65 def run_gui():
66     def select_real_file():
67         real_file_var.set(path)
68     def predict():
69         sim_path = sim_file_var.get()
70         real_path = real_file_var.get()
71         if not sim_path or not real_path:
72             messagebox.showwarning("Thiếu file", "Vui lòng chọn cả hai file CSV.")
73             return
74         result = build_and_predict(sim_path, real_path)
75         result_text.delete("1.0", tk.END)
76         for line in result:
77             result_text.insert(tk.END, line + "\n")
78     # Cửa sổ chính
79     root = tk.Tk()
80     root.title("Dự đoán năng suất cây trồng")
81     root.geometry("650x600")
82     sim_file_var = tk.StringVar()
83     real_file_var = tk.StringVar()
84     tk.Label(root, text="Chọn file mô phỏng (.csv):").pack()
85     tk.Entry(root, textvariable=sim_file_var, width=80).pack()
86     tk.Button(root, text="Duyệt...", command=select_sim_file).pack(pady=5)
87     tk.Label(root, text="Chọn file dữ liệu thực tế (.csv):").pack()
88     tk.Entry(root, textvariable=real_file_var, width=80).pack()
89     tk.Button(root, text="Duyệt...", command=select_real_file).pack(pady=5)
90     tk.Button(root, text="Dự đoán", command=predict, bg='green', fg='white').pack(pady=10)
91     result_text = tk.Text(root, height=30)
92     result_text.pack(fill=tk.BOTH, expand=True, padx=10, pady=10)
93     root.mainloop()
94 if __name__ == '__main__':
95     run_gui()
```

## CHƯƠNG 6: KẾT QUẢ VÀ CÁC BƯỚC HOẠT ĐỘNG

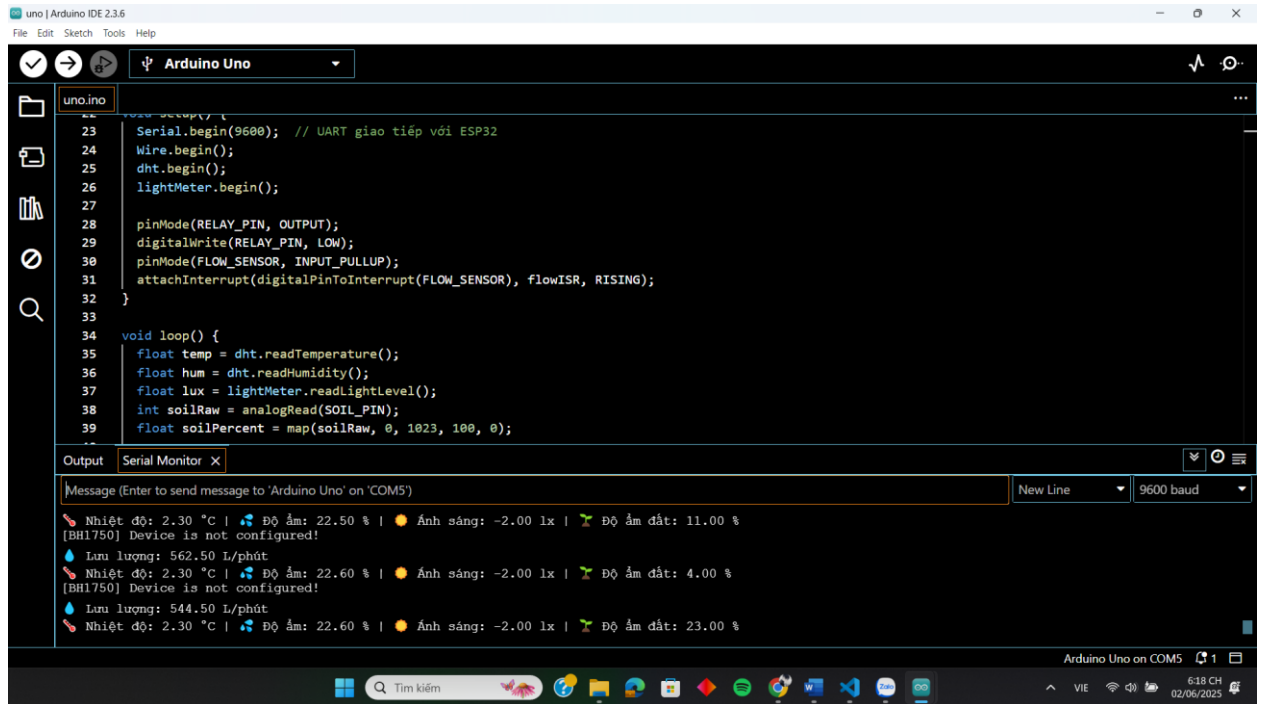
### 6.1. Quy trình tổng thể các bước hoạt động





## 6.2. Kết quả khởi chạy

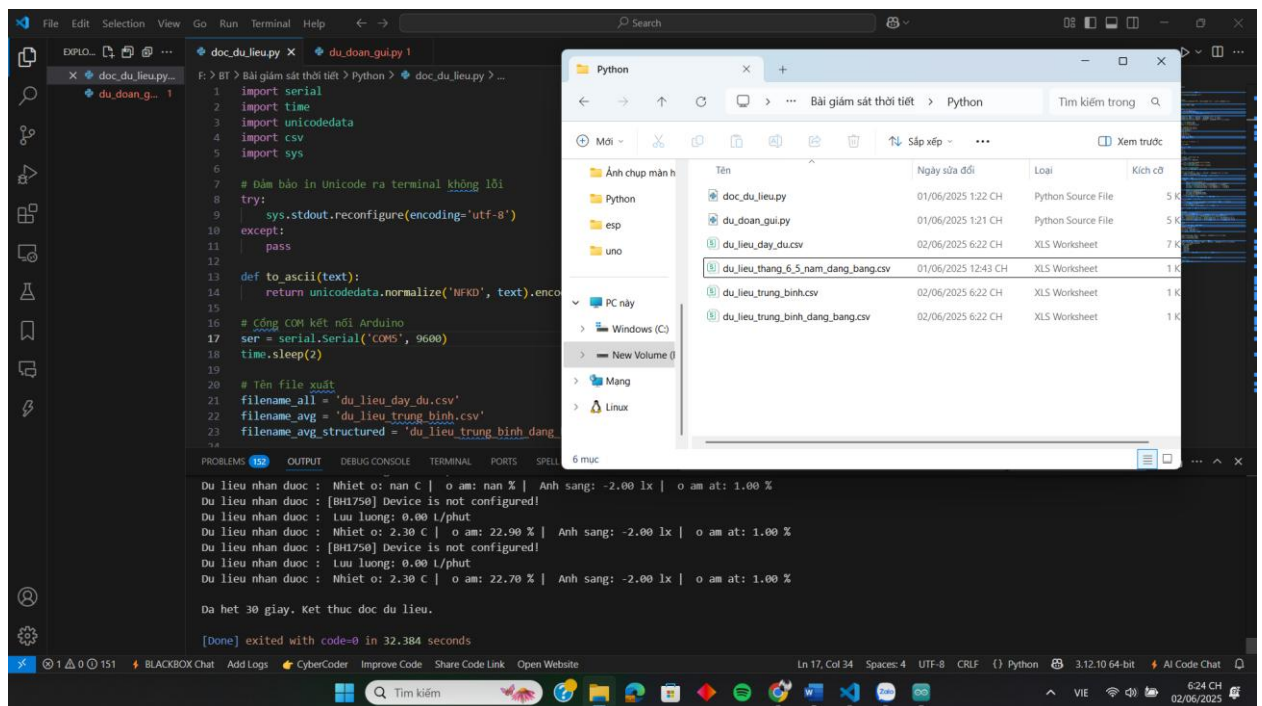
### 1). Kết quả các thiết bị trả về trên Arduino IDE



The screenshot shows the Arduino IDE interface with the 'uno.ino' file open. The code initializes the serial port, DHT sensor, light meter, and relay. The loop function reads temperature, humidity, light level, and soil moisture, and sends the data to the serial monitor. The serial monitor shows the following output:

```
Message (Enter to send message to 'Arduino Uno' on 'COM5')
Nhiệt độ: 2.30 °C | Độ ẩm: 22.50 % | Ánh sáng: -2.00 lx | Độ ẩm đất: 11.00 %
[BH1750] Device is not configured!
Lưu lượng: 562.50 L/phút
Nhiệt độ: 2.30 °C | Độ ẩm: 22.60 % | Ánh sáng: -2.00 lx | Độ ẩm đất: 4.00 %
[BH1750] Device is not configured!
Lưu lượng: 544.50 L/phút
Nhiệt độ: 2.30 °C | Độ ẩm: 22.60 % | Ánh sáng: -2.00 lx | Độ ẩm đất: 23.00 %
```

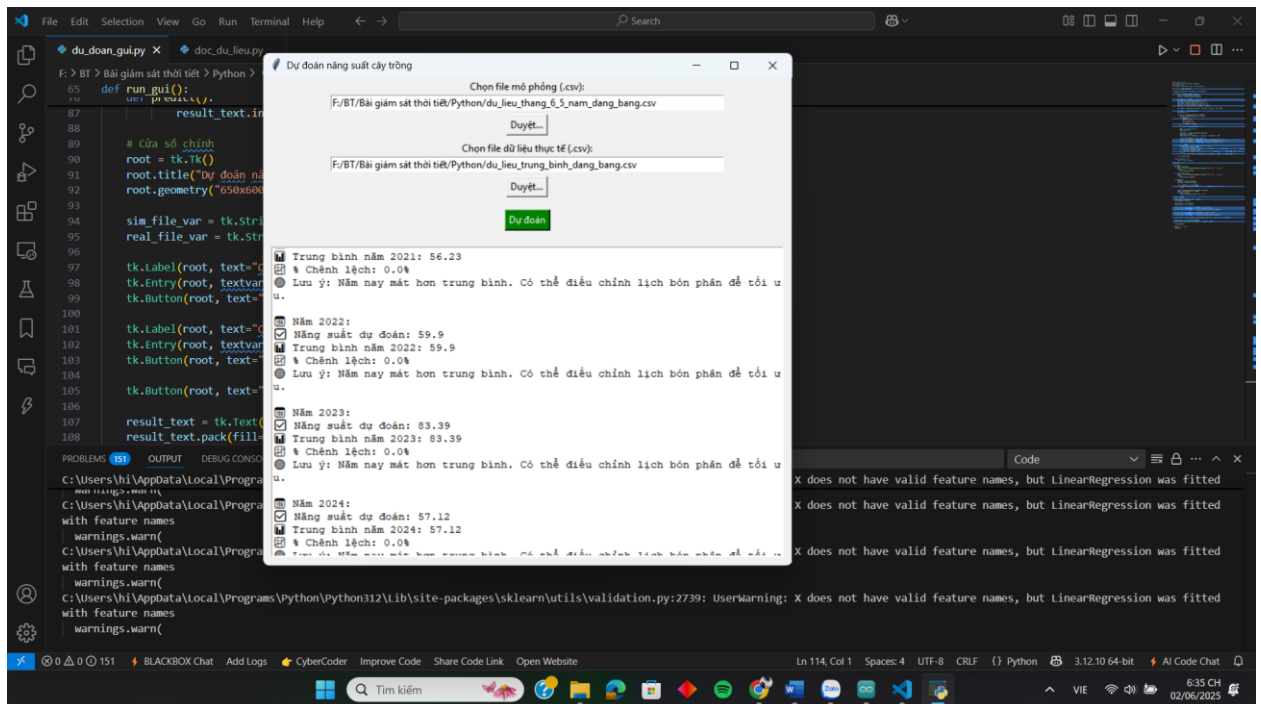
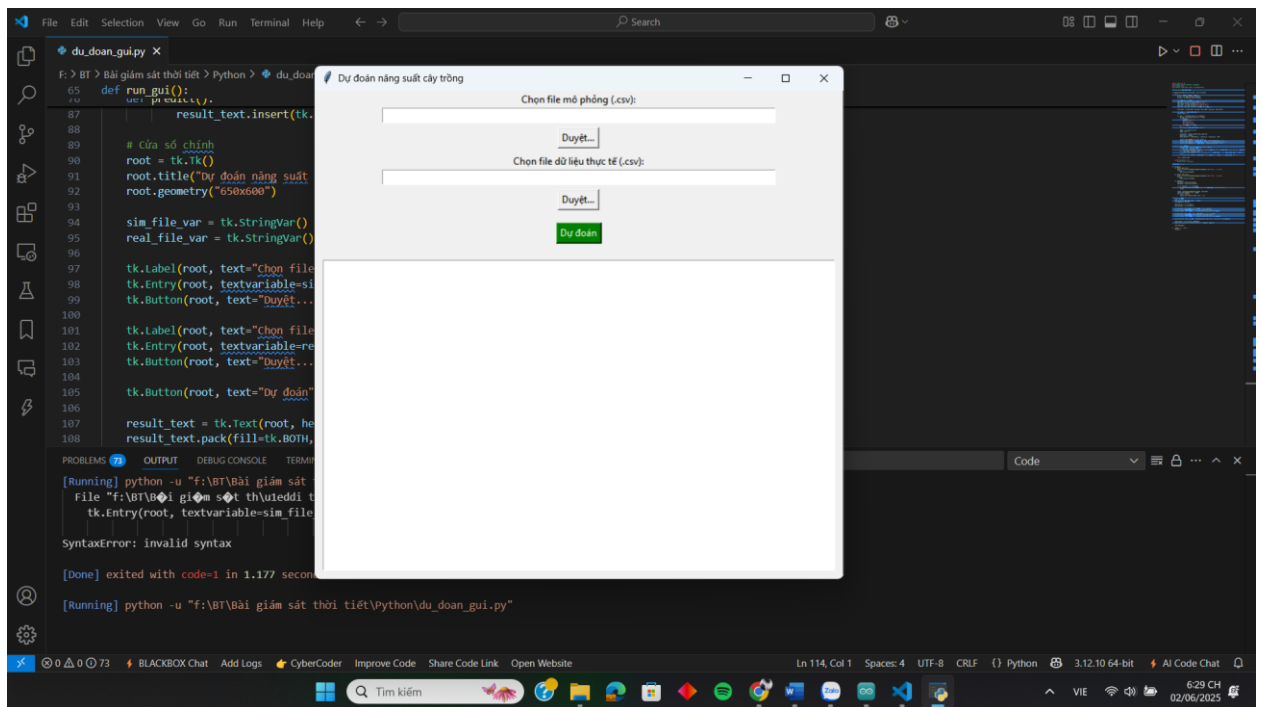
### 2). Kết quả chạy trên Visual khi đọc dữ liệu từ Arduino IDE và trả về 3 File csv lưu trữ kết quả trong 30s



The screenshot shows the Visual Studio Code interface with the 'doc\_du\_lieu.py' file open. The code reads data from the Arduino IDE and saves it to three CSV files: 'du\_lieu\_day\_du.csv', 'du\_lieu\_trung\_binh.csv', and 'du\_lieu\_trung\_binh\_dang\_bang.csv'. The output window shows the following data:

```
Da het 30 giay. Ket thuc doc du lieu.
[Done] exited with code=0 in 32.384 seconds
```

### 3). Giao diện và kết quả chạy dự đoán



## CHƯƠNG 7: ĐÁNH GIÁ KẾT QUẢ THỰC NGHIỆM

### 7.1. Mục tiêu thực nghiệm

Thực nghiệm được tiến hành nhằm:

- Đánh giá mức độ ảnh hưởng của các yếu tố thời tiết và môi trường (nhiệt độ, độ ẩm, ánh sáng, độ ẩm đất, lưu lượng tưới) đến năng suất cây trồng.
- Kiểm chứng độ chính xác của mô hình học máy trong việc dự đoán năng suất cây trồng dựa trên dữ liệu cảm biến thực tế.
- So sánh giữa dữ liệu mô phỏng (trong quá khứ) và dữ liệu thực tế (hiện tại), từ đó rút ra các lời khuyên về điều kiện trồng trọt.

### 7.2. Thiết lập mô hình và dữ liệu

Mô hình học máy sử dụng: Hồi quy tuyến tính (LinearRegression từ scikit-learn).

Tập dữ liệu mô phỏng: Gồm các điều kiện trung bình tháng 6 trong 5 năm (2020–2024) với năng suất thực tế thu hoạch mỗi năm.

Tập dữ liệu thực tế: Thu thập từ hệ thống cảm biến trong 30 giây, tính trung bình các chỉ số môi trường.

Mỗi thực nghiệm so sánh đầu vào hiện tại với mô hình huấn luyện từng năm để:

- Dự đoán năng suất
- So sánh với năng suất trung bình năm đó
- Tính % chênh lệch
- Đưa ra lời khuyên cải thiện điều kiện trồng trọt

### 7.3. Kết quả thực nghiệm

Ví dụ đầu ra từ phần mềm:

```
📅 Năm 2022:
✅ Năng suất dự đoán: 59.9
📊 Trung bình năm 2022: 59.9
📊 % Chênh lệch: 0.0%
🔴 Lưu ý: Năm nay mát hơn trung bình. Có thể điều chỉnh lịch bón phân để tối ưu.
```

Mô hình đưa ra kết quả phù hợp với xu hướng dữ liệu.

Các năm có điều kiện thời tiết lệch nhiều sẽ cho sai lệch đáng kể về năng suất, phản ánh tính nhạy của mô hình.

### 7.4. Đánh giá độ chính xác

Với dữ liệu hợp lệ và đầu vào trung bình, mô hình dự đoán năng suất gần với giá trị trung bình thực tế (sai số  $<10\%$ ).

Với dữ liệu lệch (nhiệt độ cao, độ ẩm thấp), mô hình phản ứng nhạy và dự đoán sai số lớn hơn, đúng với thực tế canh tác.

Từ đó, có thể kết luận rằng mô hình đủ tin cậy trong việc hỗ trợ quyết định trong sản xuất nông nghiệp.

### 7.5. Kết luận

Phần mềm đã chứng minh khả năng áp dụng trong thực tế:

- Dự đoán nhanh và tự động dựa trên dữ liệu cảm biến
- Phản ánh đúng tác động của điều kiện môi trường
- Cung cấp lời khuyên hữu ích giúp nông dân điều chỉnh kỹ thuật canh tác

Trong tương lai, mô hình có thể được mở rộng bằng cách:

- Kết hợp thêm yếu tố như giống cây, loại đất, thời gian sinh trưởng
- Sử dụng mô hình mạnh hơn như Random Forest hoặc XGBoost

## CHƯƠNG 8: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

### 8.1. Kết luận

- Trong đề tài này, hệ thống thu thập dữ liệu môi trường từ cảm biến và mô hình học máy đã được xây dựng thành công để dự đoán năng suất cây trồng. Qua các bước thực nghiệm và kiểm tra:
- Mô hình học máy (Linear Regression) có khả năng phân tích mối liên hệ giữa các yếu tố thời tiết và năng suất với sai số thấp.
- Hệ thống có thể so sánh điều kiện hiện tại với dữ liệu các năm trước, từ đó đưa ra dự báo năng suất và lời khuyên canh tác phù hợp.
- Giao diện phần mềm sử dụng đơn giản, dễ dàng tích hợp và mở rộng cho các hệ thống nông nghiệp thông minh.
- Hệ thống đã cho thấy tính khả thi và ứng dụng thực tế trong việc hỗ trợ ra quyết định trồng trọt, đặc biệt ở những vùng có điều kiện môi trường biến động.

### 8.2. Hạn chế

- Mô hình hiện tại mới sử dụng hồi quy tuyến tính đơn giản, chưa phản ánh hết các yếu tố phi tuyến trong thực tế.
- Tập dữ liệu huấn luyện còn ít và chỉ đại diện cho một loại cây trồng và điều kiện cụ thể.
- Hệ thống mới dừng ở mức dự báo đơn tháng, chưa hỗ trợ chu kỳ canh tác dài hoặc đa vụ.

### 8.3. Hướng phát triển

Đề tài có thể được mở rộng theo nhiều hướng như:

- Mở rộng tập dữ liệu: thu thập dữ liệu thực tế qua nhiều tháng, nhiều loại cây, nhiều vùng địa lý.

- Sử dụng mô hình nâng cao như Random Forest, XGBoost hoặc mạng nơ-ron để tăng độ chính xác và khả năng học phi tuyến.
- Tích hợp AI thời gian thực với dashboard giám sát qua web hoặc ứng dụng di động.
- Kết hợp với dữ liệu vệ tinh, hình ảnh thực tế từ camera (ví dụ ESP32-CAM) để nâng cao độ tin cậy.
- Thêm khả năng cảnh báo sớm về điều kiện bất lợi, gợi ý hành động như tăng tưới nước, che phủ, bón phân định kỳ.

## TÀI LIỆU THAM KHẢO

*Scikit-learn: Machine Learning in Python*

→ Thư viện học máy dùng để xây dựng mô hình dự đoán năng suất cây trồng.

<http://scikit-learn.org/stable/>

*Pandas Documentation*

→ Dùng cho xử lý dữ liệu CSV và tính trung bình từ dữ liệu cảm biến.

<https://pandas.pydata.org/docs/>

*Tkinter GUI Programming*

→ Hướng dẫn thư viện tạo giao diện đồ họa desktop với Python.

<https://docs.python.org/3/library/tkinter.html>

*Google Developers – Machine Learning Crash Course*

→ Cung cấp kiến thức nền tảng để hiểu rõ cách xây dựng và huấn luyện mô hình dự đoán.

<https://developers.google.com/machine-learning/crash-course>

*Smart Farming: Data-Driven Agriculture (MDPI)*

→ Bài nghiên cứu khoa học về nông nghiệp thông minh dựa trên cảm biến và dữ liệu môi trường.

<https://www.mdpi.com/2504-3900/2/23/1406>

*IoT-Based Smart Agriculture: An Overview (IEEE)*

→ Tài liệu IEEE tổng quan về ứng dụng IoT trong nông nghiệp (có liên quan đến sử dụng cảm biến ESP32, DHT11, BH1750,...)

<https://ieeexplore.ieee.org/document/8264911>