

BỘ CÔNG THƯƠNG  
TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP THÀNH PHỐ HỒ CHÍ MINH  
KHOA CÔNG NGHỆ ĐIỆN TỬ



TRẦN CÔNG HÒA  
NGUYỄN TRÍ THÚC

XÂY DỰNG MẠNG LƯỚI TRẠM KHÍ HẬU  
TRONG GIÁM SÁT NHIỄM MẶN ỦNG DỤNG  
AIOT

Chuyên ngành: IOT VÀ TRÍ TUỆ NHÂN TẠO ỦNG DỤNG  
Mã chuyên ngành: 7510304

ĐỀ CƯƠNG KHOÁ LUẬN TỐT NGHIỆP

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2025

## **THÔNG TIN CHUNG**

Họ và tên sinh viên: Trần Công Hòa..... MSSV: 20017691 .....

Lớp : DHIOT16B ..... Khóa: K16 .....

Chuyên ngành : IOT và trí tuệ nhân tạo ứng dụng Mã chuyên ngành: 7510304.

SĐT : 0869209724 .....

Email : tranconghoa24@gmail.com.....

Địa chỉ liên hệ : 129 Nguyễn Văn Nghi, Phường 7, Gò Vấp, TP. Hồ Chí Minh

Tên đề tài : Xây dựng mạng lưới trạm khí hậu trong giám sát nhiễm mặn  
ứng dụng AIOT

Người hướng dẫn : TS. Cao Văn Kiên.....

SĐT : .....

Email : .....

Cơ quan công tác : Khoa Công Nghệ Điện Tử, ĐH Công Nghiệp TPHCM

*Tp. Hồ Chí Minh, ngày .... tháng .... năm 2025*

**Người hướng dẫn**  
*(Ký và ghi rõ họ tên)*

**Sinh viên**  
*(Ký và ghi rõ họ tên)*

## **THÔNG TIN CHUNG**

Họ và tên sinh viên: Nguyễn Trí Thúc..... MSSV: 20029891 .....

Lớp : DHIOT16A..... Khóa: K16.....

Chuyên ngành : IOT và trí tuệ nhân tạo ứng dụng Mã chuyên ngành: 7510304...

SĐT : 0338494041 .....

Email : trithuc2001stvn@gmail.com .....

Địa chỉ liên hệ : 274/27 Nguyễn Văn Nghi, Phường 7, Gò Vấp, TP. Hồ Chí Minh .....

Tên đề tài : Xây dựng mạng lưới trạm khí hậu trong giám sát nhiễm mặn ứng dụng AIOT

Người hướng dẫn : TS. Cao Văn Kiên.....

SĐT : .....

Email : .....

Cơ quan công tác : Khoa Công Nghệ Điện Tử, ĐH Công Nghiệp TPHCM

*Tp. Hồ Chí Minh, ngày .... tháng ... năm 2025*

**Người hướng dẫn**

*(Ký và ghi rõ họ tên)*

**Sinh viên**

*(Ký và ghi rõ họ tên)*

## NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN:

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

Ngày ..... tháng ..... năm .....

**GIÁO VIÊN HƯỚNG DẪN**

(Ký và ghi rõ họ tên)

## **NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN :**

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

Ngày ..... tháng ..... năm .....

**GIÁO VIÊN PHẢN BIỆN**

(Ký và ghi rõ họ tên)

## **LỜI CẢM ƠN**

Nhóm xin cảm ơn và bày tỏ lòng biết ơn sâu sắc đến Thầy – TS. Cao Văn Kiên, người đồng hành với nhóm trong thời gian vừa qua. Cảm ơn thầy đã dành nhiều thời gian quý báu để hỗ trợ nhóm, truyền đạt, hướng dẫn nhóm trong quá trình thực hiện luận văn.

Nhóm xin gửi lời cảm ơn chân thành đến thầy cô trong Khoa Công Nghệ Điện Tử. Hỗ trợ, tạo điều kiện thuận lợi trong môi trường học thuật suốt thời gian học tại đây.

Ngoài ra, nhóm cảm ơn các bạn bè, gia đình luôn động viên, khích lệ và hỗ trợ về mặt tinh thần đến cả vật chất, để nhóm có thể vượt qua mọi khó khăn trong thời gian qua.

Cuối cùng, xin cảm ơn đến Công ty cổ phần Gcalls, đã hỗ trợ về chi phí cho thiết bị, và hỗ trợ về kiến thức xây dựng nên sản phẩm này.

Bên cạnh những điều đã đạt được, thì nhóm không tránh khỏi những thiếu sót. Nhóm mong muốn nhận được sự góp ý của tất cả mọi người để hoàn thiện hơn cho sau này.

## LỜI CAM ĐOAN

Nhóm xin cam đoan, nhóm tự thực hiện, tham khảo các tài liệu từ các nguồn trước đó. Đề tài “**Xây dựng mạng lưới trạm khí hậu trong giám sát nhiễm mặn ứng dụng AIOT**” được sự hỗ trợ từ Thầy – TS. Cao Văn Kiên hoàn toàn không sao chép bất cứ tài liệu hay công trình nghiên cứu nào.

*Tp. Hồ Chí Minh, ngày ... tháng ... năm 2025*

**Sinh viên**

*(Ký và ghi rõ họ tên)*

## MỤC LỤC

<b>DANH MỤC HÌNH ẢNH.....</b>	<b>1</b>
<b>DANH MỤC BẢNG BIỂU.....</b>	<b>5</b>
<b>DANH MỤC TỪ VIẾT TẮT .....</b>	<b>6</b>
<b>MỞ ĐẦU .....</b>	<b>5</b>
<b>CHƯƠNG 1      TỔNG QUAN VỀ ĐỀ TÀI .....</b>	<b>8</b>
1.1    Trạm khí hậu là gì ?.....	8
1.2    Tổng quan về IOT .....	9
1.2.1    Định nghĩa.....	9
1.2.2    IOT ứng dụng trong các lĩnh vực .....	9
1.3    Tổng quan về AI.....	10
1.3.1    Định nghĩa.....	10
1.3.2    AI được sử dụng như thế nào?.....	11
<b>CHƯƠNG 2      CƠ SỞ LÝ THUYẾT .....</b>	<b>12</b>
2.1    Các loại mạng không dây .....	12
2.1.1    Wifi .....	12
2.1.2    4G .....	12
2.1.3    Lora .....	13
2.1.4    Vai trò LoRa trong IOT .....	14
2.2    Lý thuyết về mạng nơ ron: .....	14
2.2.1    Khái niệm .....	14
2.2.2    Cấu trúc cơ bản của một mạng nơ-ron.....	15
2.2.3    Hoạt động của một nơ-ron .....	15
2.2.4    Một số hàm kích hoạt phổ biến.....	16
2.2.5    Quá trình học của mạng nơ ron.....	17
2.2.6    Ứng dụng của mạng nơ ron.....	17
2.3    Giới thiệu phần cứng .....	18
2.3.1    Giới thiệu STM32F103C8T6.....	18
2.3.2    Giới thiệu ESP32 .....	20

2.3.3	Các loại cảm biến sử dụng .....	21
2.3.4	Giới thiệu LCD 16x2 tích hợp I2C .....	25
2.3.5	Module LoRa SX1278 Ra-02 433MHz .....	26
2.3.6	Module Sim và usb sim.....	28
2.4	Các chuẩn truyền dữ liệu.....	29
2.4.1	SPI (Serial Peripheral Interface) .....	29
2.4.2	UART (Universal asynchronous receiver transmitter) .....	30
2.4.3	I2C (Inter-Integrated Circuit).....	32
<b>CHƯƠNG 3</b>	<b>PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG.....</b>	<b>35</b>
3.1	Phân tích hệ thống .....	35
3.1.1	Yêu cầu hệ thống .....	35
3.1.2	Sơ đồ khái niệm .....	36
3.2	Thiết kế hệ thống .....	37
3.2.1	Phần cứng.....	37
3.2.2	Phần mềm.....	44
3.2.3	Xây dựng mô hình AI dự đoán các giá trị trong tương lai.....	53
<b>CHƯƠNG 4</b>	<b>THỰC NGHIỆM VÀ ĐÁNH GIÁ KẾT QUẢ.....</b>	<b>67</b>
4.1	Kết quả phần cứng.....	67
4.2	Lắp đặt thực tế .....	69
4.3	Kết quả phần mềm.....	70
4.3.1	Server .....	70
4.3.2	App.....	71
4.4	Đánh giá kết quả.....	74
<b>KẾT LUẬN VÀ KIẾN NGHỊ .....</b>	<b>75</b>	
4.1	Kết luận .....	75
4.2	Kiến nghị .....	76

## **DANH MỤC HÌNH ẢNH**

Hình 1.1 Trạm khí hậu .....	8
Hình 1.2 Tổng quan IOT .....	9
Hình 1.3 Ứng dụng IOT trong các lĩnh vực .....	10
Hình 1.4 AI.....	11
Hình 2.1 Wifi .....	12
Hình 2.2 4G .....	12
Hình 2.3 Cấu trúc gói tin LoRa.....	13
Hình 3.1 Sơ đồ khối hệ thống .....	36
Hình 3.2 Sơ đồ khối Gateway .....	36
Hình 3.3 Sơ đồ khối Node.....	37
Hình 3.4 Schematic nguồn cho các cảm biến .....	37
Hình 3.5 Schematic nguồn cho STM32F103C8T6.....	38
Hình 3.6 Schematic STM32F103C8T6 ra chân .....	38
Hình 3.7 Schematic LoRa trên board Node .....	39
Hình 3.8 Schematic ra chân kết nối cảm biến lưu lượng .....	39
Hình 3.9 Schematic ra chân kết nối cảm biến SHT30 .....	40
Hình 3.10 Schematic ra chân kết nối cảm biến độ mặn .....	40
Hình 3.11 Schematic đọc kit thời tiết.....	40
Hình 3.12 Schematic đọc cảm biến 4-20mA .....	41
Hình 3.13 Thiết kế layout cho mạch node .....	41
Hình 3.14 Schematic nguồn cho module sim .....	41
Hình 3.15 Schematic nguồn cho module sim .....	42

Hình 3.16 Schematic ESP32 .....	42
Hình 3.17 Schematic LoRa trên board gateway .....	43
Hình 3.18 Schematic chân kết nối module sim.....	43
Hình 3.19 Thiết kế layout cho mạch gateway.....	44
Hình 3.20 Lưu đồ chương trình chính Node .....	45
Hình 3.21 Lưu đồ chương trình ngắn điều khiển các task .....	46
Hình 3.22 Lưu đồ chương trình task gửi dữ liệu.....	47
Hình 3.23 Lưu đồ chương trình task đọc dữ liệu từ cảm biến độ mặn .....	48
Hình 3.24 Lưu đồ chương trình task đọc lưu lượng nước .....	48
Hình 3.25 Lưu đồ chương trình chính gateway .....	49
Hình 3.26 Lưu đồ chương trình ngắn gateway .....	50
Hình 3.27 Giao tiếp giữa node và Gateway .....	51
Hình 3.28 Các luồng hoạt động của app .....	52
Hình 3.29 Lấy dữ liệu từ Thingspeak và tiến hành phân tích .....	54
Hình 3.30 Phân tích dữ liệu.....	54
Hình 3.31 Tiến xử lí dữ liệu.....	55
Hình 3.32 Trực quan hóa bằng biểu đồ .....	55
Hình 3.33 Sau khi xử lí nhiễu .....	56
Hình 3.34 Phân tách và chuẩn hóa dữ liệu.....	57
Hình 3.35 Cửa sổ với 1 đầu ra dựa trên 6 giá trị lịch sử .....	58
Hình 3.36 Cửa sổ sử dụng cho mô hình.....	58
Hình 3.37 Lớp WindowGenerator .....	59
Hình 3.38 Cấu trúc mô hình.....	60

Hình 3.39 Khởi tạo mô hình.....	60
Hình 3.40 Kiến trúc Mô hình LSTM .....	61
Hình 3.41 Huấn luyện lần 1 .....	62
Hình 3.42 Huấn luyện lần 2 .....	62
Hình 3.43 Huấn luyện lần 3 .....	62
Hình 3. 3.444 Kết quả kiểm tra với đặc trưng độ mặn.....	63
Hình 3.45 Kiến trúc mô hình LSTM sau khi cải tiến.....	64
Hình 3.46 Huấn luyện sau khi cải tiến .....	64
Hình 3.47 Kết quả kiểm tra với đặc trưng độ mặn sau khi cải tiến .....	65
Hình 3. 3.48 Lưu đồ chương trình AI dự đoán tương lai.....	66
Hình 4.1 Hàn linh kiện lên mạch in .....	67
Hình 4.2 Test thiết bị sau lúc đóng hộp .....	67
Hình 4.3 Đóng gói các mạch vào hộp chống nước .....	68
Hình 4.4 Gateway sau khi đóng gói .....	68
Hình 4.5 Node sau khi đóng gói.....	69
Hình 4.6 Hình ảnh lắp đặt thực tế .....	69
Hình 4.7 Hình ảnh dữ liệu được lấy được từ thiết bị IOT .....	70
Hình 4.8 Giao diện login của app.....	71
Hình 4.9 Giao diện khi thông tin đăng nhập sai.....	71
Hình 4.10 Quản lý các nông trại .....	72
Hình 4.11 Các giá trị hiện tại của cảm biến .....	72
Hình 4.12 Biểu đồ giá trị của nhiệt độ .....	73
Hình 4.13 Biểu đồ giá trị của độ ẩm .....	73

Hình 4.14 Biểu đồ giá trị của độ mặn ..... 73

Hình 4.15 Biểu đồ giá trị sau khi dự đoán ..... 74

## **DANH MỤC BẢNG BIỂU**

Bảng 2.1 Các chân SHT30 .....	22
Bảng 2.2 Các chân cảm biến lưu lượng nước .....	23
Bảng 2.3 Các chân cảm biến độ mặn .....	24
Bảng 2.4 Các chân của board chuyển đổi EC .....	24
Bảng 2.5 Các chân Board chuyển đổi nhiệt độ .....	25
Bảng 2.6 Các chân của module SX1278 .....	27

## **DANH MỤC TỪ VIẾT TẮT**

<b>MOSI</b>	Master gửi dữ liệu đến Slave.
<b>MISO</b>	Slave gửi dữ liệu cho Master.
<b>SCLK</b>	Xung giữ nhịp Clock.
<b>BW</b>	Băng thông
<b>SF</b>	Hệ số trải phổ
<b>CR</b>	Tốc độ mã hóa
<b>IOT</b>	Internet of Things
<b>LoRa</b>	Long Range
<b>SPI</b>	Serial Peripheral Interface
<b>I2C</b>	Inter – Integrated Circuit
<b>UART</b>	Universal Asynchronous Receiver-Transmitter
<b>MCU</b>	Micro controller Unit
<b>SCL</b>	Serial Clock Line
<b>SDA</b>	Serial Data Line
<b>CRC</b>	Cyclic Redundancy Check
<b>EC</b>	Độ dẫn điện

## MỞ ĐẦU

### 1. Đặt vấn đề

Nông nghiệp quả thực đóng vai trò vô cùng quan trọng trong nền kinh tế Việt Nam. Với hàng triệu người sống dựa vào nông nghiệp, đất nước này đã trở thành một trong những nhà sản xuất hàng đầu về gạo, cà phê, ... Nông nghiệp không chỉ giúp ổn định đời sống của người dân mà còn đóng góp lớn vào xuất khẩu và tăng trưởng kinh tế. Cụ thể, lĩnh vực nông, lâm, ngư nghiệp chiếm 11,96% sản phẩm quốc nội (GDP) [1]. Trong lĩnh vực xuất khẩu, giá trị nông, lâm, ngư sản đạt 53,01 tỷ USD trong năm 2023, và Việt Nam giữ vị trí cao cụ thể là vị trí thứ 3 của về việc xuất khẩu gạo [2]. Nguồn nước ngọt là tài nguyên thiết yếu dành cho nông nghiệp. Nông nghiệp cần tới 82% tổng lượng nước dành cho tưới tiêu, lên tới 66.000 triệu m<sup>3</sup>/năm (năm 2018) [3]. Nhưng hiện nay xâm nhập mặn đan đe dọa trực tiếp tới nguồn nước ở các cửa sông. Biến đổi khí hậu làm gia tăng xâm nhập mặn, nước biển dâng cao khiến cho xâm nhập sâu ăn vào đất liền làm ảnh hưởng trực tiếp đến sản xuất nông nghiệp vùng ven biển. Ngoài ra còn làm xảy ra thường xuyên, như hạn hán, bão, lũ cũng 1 phần gây ra tăng độ mặn cho các vùng nước ngọt.

Nguồn nước ngọt bị ngập mặn ảnh hưởng trực tiếp đến nguồn nước ngọt, đe dọa an toàn trực tiếp đến nguồn thức ăn và tăng trưởng của nông nghiệp.

Tác động đến nền nông nghiệp:

- Cây lúa, cây ăn trái bị chết do thiếu nước, nhiễm mặn làm giảm năng suất và sản lượng gây ảnh hưởng đến cuộc sống và nguồn thức ăn của quốc gia.
- Gây thoái hóa đất, suy thoái hệ sinh thái, tác động mạnh đến thảm thực vật.

Tác động đến sinh hoạt:

- Nước mặn xâm nhập làm ô nhiễm nhiều nguồn nước ngọt như nước giếng, nước sông ngòi.
- Chi phí xử lý nước mặn tăng cao, gây áp lực lên ngân sách địa phương.

Tác động đến vùng ven biển:

- Các khu rừng ngập mặn các vùng ven biển bị chết, thảm thực vật ven biển sẽ không còn đa dạng, ảnh hưởng đến sự đa dạng của các loài sinh vật, ngành du lịch và ngành khai thác thủy sản.

Theo nhận định từ Viện Khoa học Thủy lợi miền Nam chỉ trong năm 2023, hiện tượng xâm nhập mặn đã ảnh hưởng đến hơn 1,5 triệu ha diện tích đất canh tác nông nghiệp tại ĐBSCL, gây ra diến biến xấu đến 10 triệu tấn lúa. Diện tích đất bị nhiễm mặn tại ĐBSCL dẫn ra nhiều hệ quả nghiêm trọng đến làm ra sản phẩm nông nghiệp và gây tác động to lớn đến bảo vệ lương thực quốc gia, ước tính đến năm 2030 có thể tăng lên đến 45% [4].

Hiện trạng thiếu hụt nước ngọt vào lúc thời tiết khô vào khoảng  $4,0 \text{ tỉ m}^3$  hàng năm đang xảy ra ở DBSCL [5].

Các phương pháp giám sát truyền thống có các giới hạn như giá trị hoạt động cao, tốn nhiều nhân lực, hiệu quả thu thập dữ liệu thấp. Khó khăn trong việc lấy các dữ liệu liên tục, thời gian thực ở vùng có địa hình khó khăn. Thiếu khả năng phân tích dữ liệu chuyên sâu để đưa ra dự báo chính xác.

## 2. Mục tiêu của đề tài

Đề tài nhằm thực hiện mục tiêu “Xây dựng mạng lưới trạm khí hậu trong giám sát nhiễm mặn ứng dụng AIOT”, thiết kế hệ thống và triển khai thi công thực tế với hệ thống bao gồm:

Gateway có khả năng giao tiếp truyền nhận dữ liệu với các node thông qua mạng LoRa, giao tiếp truyền nhận với máy chủ thông qua giao thức MQTT. Các node có khả năng lấy dữ liệu từ các cảm biến, thực hiện truyền nhận đến các node lân cận và Gateway thông qua LoRa.

Phần mềm sẽ có ứng dụng xem theo dõi cho người sử dụng. Dữ liệu được thu thập từ thiết bị cảm biến có thể giúp các chuyên gia trong việc nghiên cứu, từ đó đưa ra lời giải pháp ứng phó với việc biến đổi thời tiết và nguồn nước nhiễm mặn.

### **3. Đối tượng và phạm vi nghiên cứu**

Đối tượng nghiên cứu của đề tài là Lora vào trong xây dựng mạng lưới trạm khí hậu trong giám sát nhiễm mặn.

#### **Phạm vi nghiên cứu:**

- Hoạt động gửi và nhận dữ liệu giữa các thiết bị IoT sử dụng công nghệ LoRa.
- Thu thập dữ liệu từ môi trường bằng các cảm biến, điều các thông số đo đúng với giá trị thực tế.
- Dự đoán các thông số môi trường trong tương lai gần dựa trên các dữ liệu đã thu thập trong quá khứ.

### **4. Ý nghĩa thực tiễn của đề tài**

- Đề tài giúp đưa ra lượng thông tin, cập nhật về tình hình nhiễm mặn và thay đổi thất thường thời tiết, giúp giảm thiểu thiệt hại gây ra do xâm nhập mặn và biến đổi khí hậu.
- Hỗ trợ công tác điều tiết nguồn nước.
- Cung cấp dữ liệu khoa học tin cậy cho các nghiên cứu về tác động của thay đổi thời tiết và ảnh hưởng của nhiễm mặn.
- Phát triển hệ thống giúp đưa ra được thông tin, cảnh báo sớm thực trạng xâm nhập mặn và các hiện tượng liên quan đến thời tiết xấu.

# CHƯƠNG 1 TỔNG QUAN VỀ ĐỀ TÀI

## 1.1 Trạm khí hậu là gì ?

Trạm khí hậu là một hệ thống chuyên dụng để đo lường và ghi nhận các thông số thời tiết, bao gồm nhiệt độ, độ ẩm không khí, lượng mưa, áp suất khí quyển, cùng tốc độ và hướng gió, nhằm lấy dữ liệu về thời tiết tại khu vực cụ thể.



Hình 1.1 Trạm khí hậu

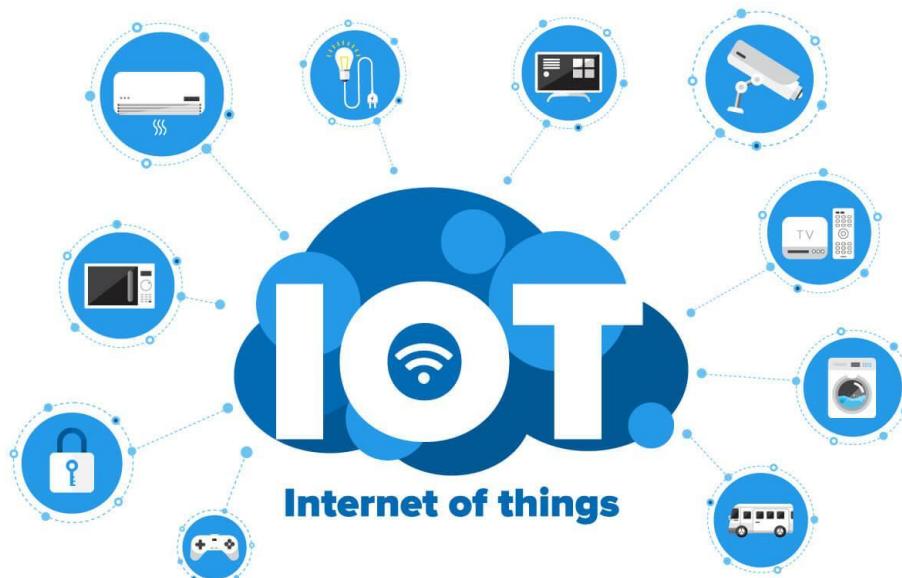
Hệ thống mạng lưới trạm khí hậu được thiết kế để ghi nhận các thông số giá trị môi trường quan trọng như điều kiện khí tượng và thủy văn, bao gồm nhiệt độ, độ ẩm và nhiều yếu tố khác. Thông qua dữ liệu này, các chuyên gia có thể theo dõi và phân tích từ xa, giúp giảm thiểu nhu cầu phải trực tiếp đến hiện trường trong quá trình giám sát.

Trạm khí hậu có vai trò rất lớn trong một số lĩnh vực, chẳng hạn nông nghiệp giúp nông dân lên kế hoạch gieo trồng hoặc trong dự báo về thiên tai, .... Không chỉ đóng vai trò thu thập thông số khí hậu, trong giai đoạn phát triển AI hiện nay thì những dữ liệu thu thập được khá quý giá để phát triển các AI dự đoán thời tiết.

## 1.2 Tổng quan về IOT

### 1.2.1 Định nghĩa

Internet of Things (IoT) đề cập đến mạng lưới các thiết bị vật lý được tích hợp cảm biến, phần mềm và công nghệ hiện đại, liên kết, truyền đạt và chia sẻ dữ liệu với các hệ thống khác qua internet. Nhiều năm gần đây, IoT đã nổi lên như một cùm từ phổ biến, cũng như là phần công nghệ quan trọng hàng đầu của thế kỷ 21. Nhờ đó, các thiết bị hữu hình hàng ngày như thiết bị phụ trợ cho, phương tiện di chuyển, thiết bị làm biến đổi nhiệt độ và thiết bị trông chừng trẻ em có thể liên kết với internet thông qua các thiết bị nhúng, tạo nên sự giao tiếp mượt mà giữa nhân loại, và các thiết bị [6].



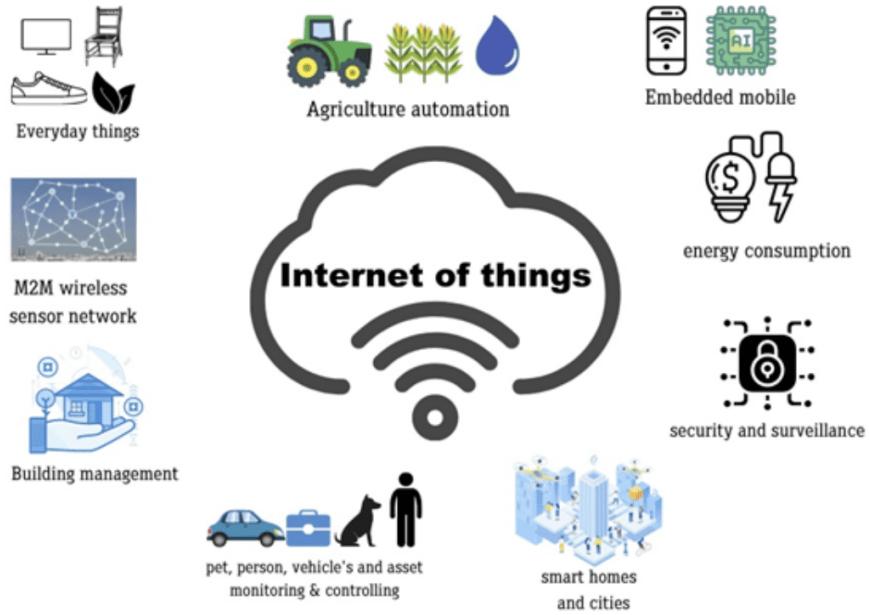
Hình 1.2 Tổng quan IOT

### 1.2.2 IOT ứng dụng trong các lĩnh vực

**Nông nghiệp:** Hệ thống các sản phẩm quan trắc, thời tiết, ....

**Công nghiệp:** Quản lý chuỗi cung ứng, bảo trì dự đoán, ....

**Y tế:** Thiết bị khám chữa bệnh từ xa, theo dõi bệnh nhân, ....



Hình 1.3 Ứng dụng IOT trong các lĩnh vực

### 1.3 Tổng quan về AI

#### 1.3.1 Định nghĩa

AI, còn được biết đến như Trí tuệ nhân tạo, là một công nghệ có khả năng giải quyết vấn đề như con người. Cách thức hoạt động của AI dường như mô phỏng trí tuệ của con người, nó có thể nhận dạng hình ảnh, làm thơ và đưa ra dự đoán dựa trên dữ liệu.

Các tổ chức phát triển hiện nay thu thập khối lượng dữ liệu cực lớn từ đa dạng các nguồn khác nhau như thiết bị hỗ trợ đo lường smart, văn bản hay hợp đồng do người tạo ra và bản ghi hệ thống. Công nghệ trí tuệ nhân tạo mô xé và tái dữ liệu để phụ trợ hoạt động kinh doanh một cách hiệu quả. Cuối cùng, mục tiêu của trí tuệ nhân tạo là làm cho phần mềm thông minh hơn để đáp ứng tương tác người dùng tùy chỉnh và giải quyết vấn đề phức tạp [7].



Hình 1.4 AI

### 1.3.2 *AI được sử dụng như thế nào?*

**Đề xuất nội dung:** Sử dụng AI để xuất hình ảnh, văn bản, âm thanh theo độ ưu thích của người dùng.

**Chăm sóc sức khỏe:** Sử dụng AI đưa ra các mặt hàng thích hợp với quan tâm.

**Dự báo:** Sử dụng AI đưa ra phán đoán sự vật hay hiện tượng trong tương lai bằng việc dựa vào dữ liệu trong lịch sử [7].

## CHƯƠNG 2 CƠ SỞ LÝ THUYẾT

### 2.1 Các loại mạng không dây

#### 2.1.1 Wifi

Wifi là một loại mạng không dây được tuân theo tiêu chuẩn “802.11b”. Trong nhiều năm qua, nó phát triển và thống trị chuẩn mạng LAN không dây. Độ bao phủ từ 100-500 feet. [8].



Hình 2.1 Wifi

#### 2.1.2 4G

4G là thế hệ mạng di động thứ tư, năm 2009 là lần phổ biến mạng này đầu tiên. Mạng 4G mang trải nghiệm tốc độ truyền đạt dữ liệu phát triển hơn. Ngày nay, hơn 50% kết nối di động được thực hiện trên mạng 4G [9].



Hình 2.2 4G

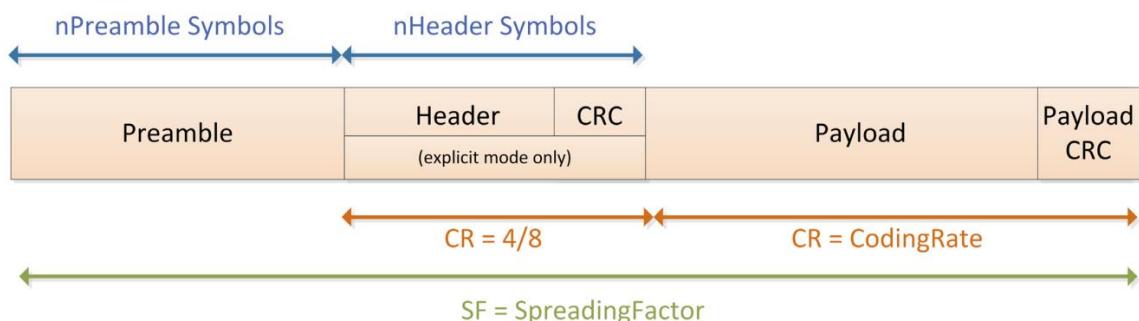
### 2.1.3 Lora

#### 2.1.3.1 Công nghệ lora

LoRa là một công nghệ truyền dữ liệu có khả năng hoạt động ở khoảng cách xa lên đến hàng chục km, đồng thời tối ưu hóa năng lượng tiêu thụ. Công nghệ này phát triển bởi Semtech và dựa trên kỹ thuật điều chế tín hiệu CSS. Nhờ sử dụng kỹ thuật này, LoRa có thể mở rộng phạm vi truyền dữ liệu trong khi ổn định mức tiêu thụ thấp về năng lượng [10].

#### 2.1.3.2 Cấu trúc gói tin

Có 2 loại gói tin là explicit và implicit. Gói explicit bao gồm một header ngắn chứa thông tin về số byte, tỷ lệ mã hóa (coding rate), và việc sử dụng CRC trong gói tin.



Hình 2.3 Cấu trúc gói tin LoRa

Các thành phần trong 1 gói tin của LoRa:

- **Preamble:** đóng vai trò trong việc giúp bộ thu đồng bộ với dòng dữ liệu đang được truyền tới.
- **Header:** Lưu trữ thông tin size của payload.
- **CRC:** phát hiện trong phần header.
- **Payload:** Bao gồm dữ liệu được truyền thông qua LoRa.
- **Payload CRC:** Lưu trữ giá trị CRC của payload. [10].

#### 2.1.3.3 Phạm vi hoạt động

Thông số ảnh hưởng phạm vi : Phạm vi hoạt động của LoRa được tác động bởi các yếu tố như CR, SF, BW. Băng thông là phần tử là những tần số ở đó tín hiệu có thể biến đổi, với các mức phổ biến là 125kHz, 250kHz, và 500kHz. Cấu hình băng thông

lớn giúp mã hóa tín hiệu và tốc độ cao khi gửi dữ liệu, nhưng lại giảm vùng truyền. Hệ số trại phổ (SF) xác định số lượng tín hiệu chirp dùng để mã hóa dữ liệu sau khi điều chế. SF dao động từ 7 đến 12, nghĩa là mỗi mức logic sẽ được mã hóa thành n xung tín hiệu, với n bằng giá trị của SF. Khi SF tăng, thời gian truyền dài hơn, tỉ lệ lỗi bit giảm, và phạm vi truyền xa hơn. Tốc độ mã hóa (CR) là số bit bổ sung trong gói tin, giúp mạch thu sửa lỗi và khôi phục gói tin khi xảy ra sai sót. [11].

Băng tần: Có 2 loại băng tần có cấp phép hoạt động (licensed band) và băng tầng không cấp phép (unlicensed). Băng tần của LoRa hoạt động ở các khu vực sẽ khác nhau, khu vực Châu Âu là 866MHz, Châu Á là 433MHz, Bắc Mỹ là 915MHz. Ở Việt Nam băng tầng 433MHz được miễn giấy phép [12].

#### **2.1.4 Vai trò LoRa trong IOT**

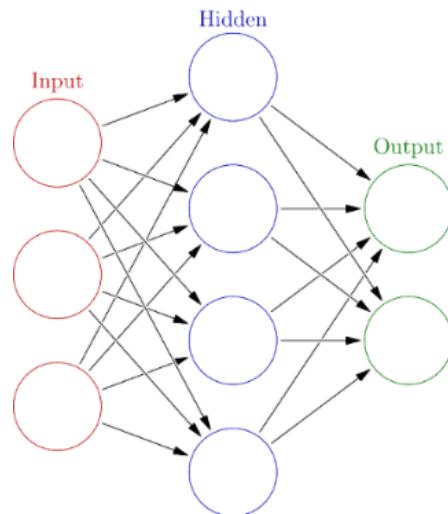
LoRa gần đây được phổ biến nhanh chóng vào IoT nhờ tính năng tiết kiệm điện xuyên suốt quá trình hoạt động. Đồng thời, gửi dữ liệu dùng trong công nghệ này ở khoảng cách xa, có thể đạt tới vài km.

### **2.2 Lý thuyết về mạng nơ ron:**

#### **2.2.1 Khái niệm**

Mạng nơ-ron nhân tạo là mô hình tính toán được lấy cảm hứng từ cách hoạt động của các nơ-ron sinh học trong bộ não của con người. Nó bao gồm nhiều nút (neurons) được liên kết với nhau bằng các trọng số (weights).

### 2.2.2 Cấu trúc cơ bản của một mạng nơ-ron



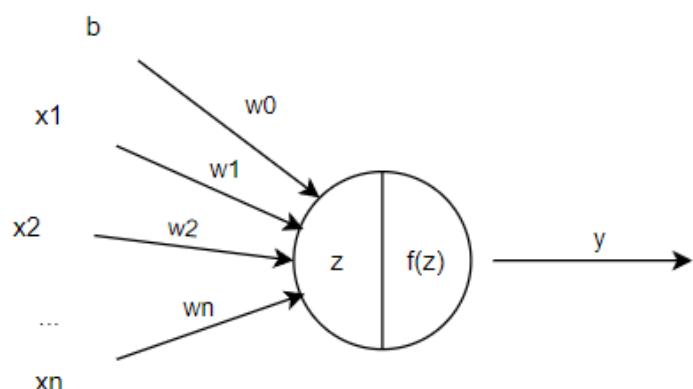
Hình 2.4 Cấu trúc mạng nơ ron

Một mạng nơ-ron cơ bản bao gồm những lớp sau:

- **Lớp đầu vào (Input layer):** Có 1 lớp và được dùng để nhận dữ liệu từ đầu vào
- **Lớp ẩn (Hidden layer):** Có thể có 1 hoặc nhiều lớp dùng để xử lý thông tin
- **Lớp đầu ra (Output layer):** Có 1 lớp dùng để làm đầu ra cho kết quả dự đoán

Trong mạng nơ-ron, mỗi nơ-ron sẽ được kết nối với các nơ-ron khác trong mạng thông qua các liên kết là các trọng số.

### 2.2.3 Hoạt động của một nơ-ron



Hình 2.5 Cấu tạo perceptron

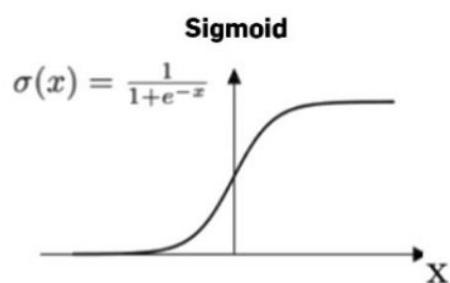
Mỗi nơ-ron sẽ tiếp nhận tín hiệu đầu vào từ lớp đầu vào hoặc từ các nơ-ron lớp trước đó, sau đó sẽ tính tổng có trọng số của các đầu vào:  $z = \sum(w_i * x_i) + b$ .

Trong đó:  $w_i$  là trọng số,  $x_i$  là đầu vào,  $b$  là độ lệch (bias)

Sau đó áp dụng hàm kích hoạt (activation function) để tạo đầu ra:  $y = f(z)$ .

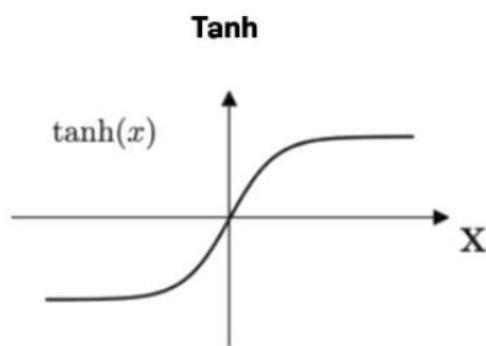
#### 2.2.4 Một số hàm kích hoạt phổ biến

- Hàm Sigmoid:  $f(x) = \frac{1}{1+e^{-x}}$



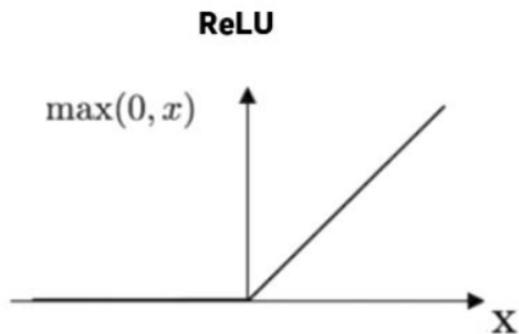
Hình 2.6 Hàm sigmoid

- Hàm Tanh:  $f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$



Hình 2.7 Hàm tanh

- Hàm ReLU:  $f(x) = \max(0, x)$



Hình 2.8 Hàm ReLU

### 2.2.5 Quá trình học của mạng nơ ron

Lan truyền xuôi (Forward propagation): Tính toán đầu ra dựa trên các đầu vào và các trọng số, sau đó tính toán sai số giữa kết quả đầu ra dự đoán và kết quả thực tế.

Lan truyền ngược (Backpropagation): Các trọng số tham gia vào quá trình tính toán sẽ được điều chỉnh để giảm thiểu sai số giữa đầu ra dự đoán với kết quả thực tế, một số thuật toán tối ưu được sử dụng để cập nhật trọng số như Gradient Descent (GD) Stochastic Gradient Descent (SGD), Momentum, Adam...

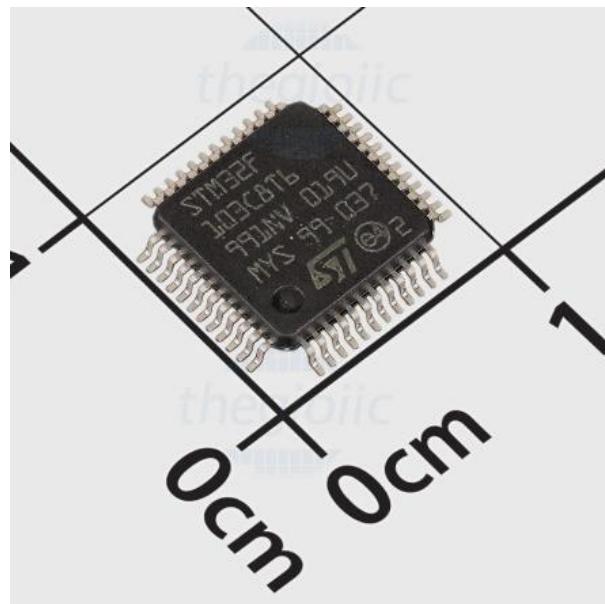
### 2.2.6 Ứng dụng của mạng nơ ron

Mạng nơ ron được ứng dụng trong nhiều lĩnh vực như:

- Nhận dạng hình ảnh
- Xử lý ngôn ngữ tự nhiên
- Dự đoán chuỗi thời gian
- Phân loại và hồi quy

## 2.3 Giới thiệu phần cứng

### 2.3.1 Giới thiệu STM32F103C8T6



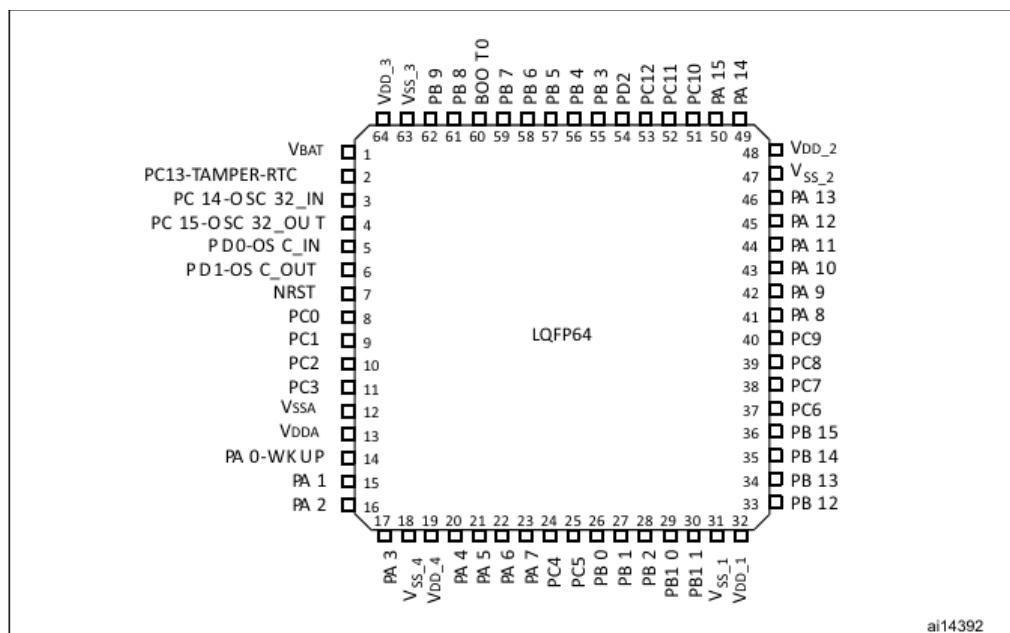
Hình 2.9 STM32F103C8T6

#### Thông số kỹ thuật:

- Kiến trúc vi điều khiển: ARM Cortex M3.
- Độ rộng của bus dữ liệu: 32-bit.
- Tần số tối đa: 72MHz.
- Điện áp hoạt động: 2V ~ 3.6V.
- Bộ nhớ Flash: 64KB.
- RAM: 20KB.
- Các chế độ hoạt động: Sleep mode, Stop mode và Standby mode.

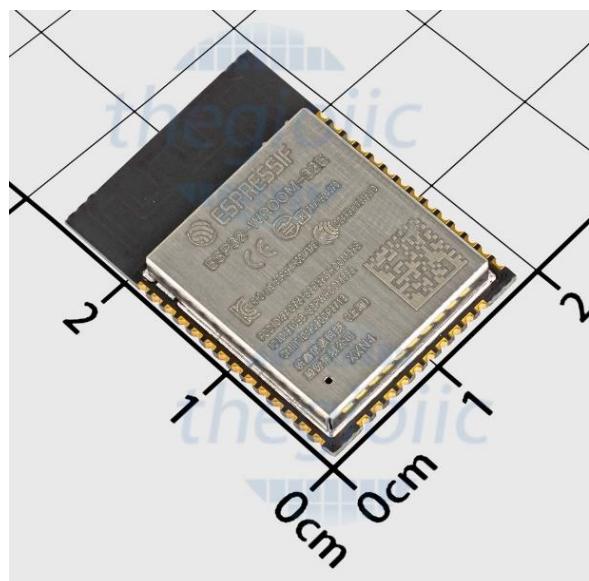
Peripheral	STM32F103Tx	STM32F103Cx	STM32F103Rx	STM32F103Vx
<b>Flash - Kbytes</b>	64	128	64	128
<b>SRAM - Kbytes</b>	20	20	20	20
Timers	General-purpose	3	3	3
	Advanced-control	1	1	1
Communication	SPI	1	2	2
	I <sup>2</sup> C	1	2	2
	USART	2	3	3
	USB	1	1	1
	CAN	1	1	1
<b>GPIOs</b>	26	37	51	80
<b>12-bit synchronized ADC</b>	2	2	2	2
<b>Number of channels</b>	10 channels	10 channels	16 channels <sup>(1)</sup>	16 channels
<b>CPU frequency</b>	72 MHz			
<b>Operating voltage</b>	2.0 to 3.6 V			
<b>Operating temperatures</b>	Ambient temperatures: -40 to +85 °C / -40 to +105 °C (see <a href="#">Table 9</a> ) Junction temperature: -40 to + 125 °C (see <a href="#">Table 9</a> )			
<b>Packages</b>	VFQFPN36	LQFP48, UFQFPN48	LQFP64, TFBGA64	LQFP100, LFBGA100, UFBGA100

Hình 2.10 Các tính năng và số lượng ngoại vi trên STM32F103xx



Hình 2.11 Sơ đồ chân STM32F103xx

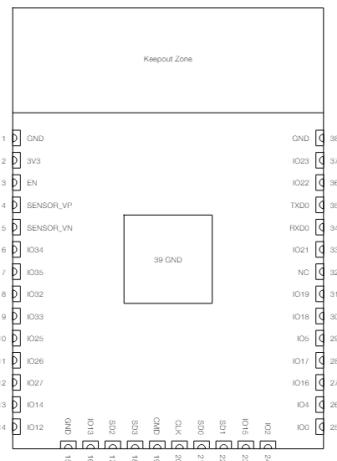
### 2.3.2 Giới thiệu ESP32



Hình 2.12 ESP32-WROOM-32D-N4

#### Thông số kỹ thuật:

- Chip: ESP32-D0WD.
- Độ rộng data bus: 32-bit.
- Tần số hoạt động: 2.4GHz.
- Điện áp hoạt động: 2.7V ~ 3.6V.
- Dòng điện hoạt động: trung bình 80mA.
- Bộ nhớ Flash: 4MB.
- RAM: 536KB
- ROM: 448KB
- Các chế độ hoạt động: Sleep mode, Stop mode và Standy mode.
- Hỗ trợ Wifi, Bluetooth,...



Hình 2.13 Sơ đồ chân ESP32-WROOM-32D-N4

### 2.3.3 Các loại cảm biến sử dụng

#### 2.3.3.1 SHT30



Hình 2.14 SHT30

#### Thông số kỹ thuật:

- Cảm biến: SHT30.
- Dải điện áp hoạt động: 2.15 đến 5.5VDC.
- Mức tín hiệu giao tiếp: TTL, từ 3.3V đến 5VDC.
- Loại giao tiếp: I2C.
- Vỏ cảm biến bằng nhựa PE chống bụi, nước.

Bảng 2.1 Các chân SHT30

Chân	Tên	Mô tả
1	VCC	Power
2	GND	Ground
3	SDA	Serial Data
4	SCL	Serial Clock

### 2.3.3.2 Cảm biến lưu lượng nước



Hình 2.5 Cảm biến lưu lượng nước

#### Thông số kỹ thuật:

- Điện áp hoạt động: 5 ~ 18VDC.
- Lưu lượng đo: 5 ~ 150 (L/min) ( $\pm 3\%$  sai số).
- Khả năng chịu được áp lực: 1.75Mpa.

### Công thức tính:

$$F = 0.45 * Q (\pm 5\% \text{ sai số})$$

**F:** Tần số (đơn vị: Hz).

**Q:** Lưu lượng nước được đo trong vòng 1 phút (đơn vị: L/min).

**0.45:** Đây là hệ số tương ứng với cảm biến nhà cung cấp đưa ra.

Bảng 2.2 Các chân cảm biến lưu lượng nước

Chân	Tên	Mô tả
1	VCC	Power
2	OUT	Tín hiệu ra
3	GND	Ground

#### 2.3.3.3 Cảm Biến Độ Dẫn Điện EC

Sử dụng cảm biến độ dẫn điện EC để đo độ mặn của nước, vì khi độ mặn tăng, khả năng dẫn điện của nước cũng sẽ tăng theo. Khi nước bị nhiễm mặn, các ion tự do trong nước giúp dẫn điện dễ dàng hơn.



Hình 2.16 Cảm biến độ dẫn điện EC

### Thông số kỹ thuật:

- Năng lượng cung cấp: 3.3 ~ 5.0V.
- Tín hiệu đầu ra: 0-3V.
- Loại đầu dò: Industrial Grade.
- Đầu dò nhiệt độ: Nhiệt kế điện trở bạch kim PT1000 (PT1000 platinum resistance thermometer).
- Phạm vi tối đa (Maximum Range): 1-2200 $\mu$ s / cm.
- Phạm vi thực tế (Practical Range): 100-2000 $\mu$ s / cm.
- Nhiệt độ hoạt động: 0-50 °C (non-freezing).
- Chịu áp lực (Pressure Resistant): 0.5Mpa.
- Mức độ chống thấm nước (Waterproof Level): IP68.

Bảng 2.3 Các chân cảm biến độ mặn

Chân	Tên	Mô tả
1	S+ (EC)	Điện cực dương
2	S- (EC)	Điện cực âm
3	TEMP	PT1000 RTD, không nhạy cảm với phân cực

Bảng 2.4 Các chân của board chuyển đổi EC

Chân	Tên	Mô tả
1	GND	Ground
2	VCC	Power

3	A	Tín hiệu ra analog
4	S+	Kết nối đầu dò dương
5	S-	Kết nối đầu dò âm

Bảng 2.5 Các chân Board chuyển đổi nhiệt độ

Chân	Tên	Mô tả
1	GND	Ground
2	VCC	Power
3	A	Tín hiệu ra dạng analog
4	T+	Kết nối đầu dò dương
5	T-	Kết nối đầu dò âm

### 2.3.4 Giới thiệu LCD 16x2 tích hợp I2C



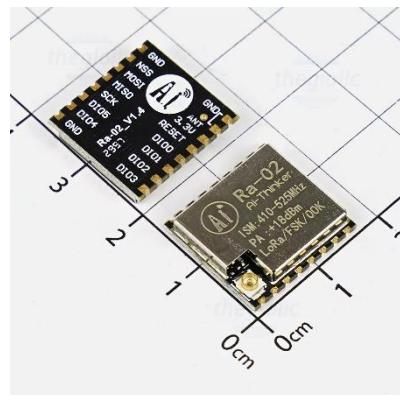
Hình 2.17 LCD tích hợp I2C

#### Thông số kỹ thuật:

- Năng lượng cung cấp: 5V.
- Chuẩn giao tiếp: I2C.
- Điện áp giao tiếp: TTL 3.3/5V.

- Loại LCD: LCD1602.
- IC chuyển giao tiếp LCD sang

### 2.3.5 Module LoRa SX1278 Ra-02 433MHz



Hình 2.18 Lora Ra-02 AI-Thinker

#### Thông số kỹ thuật:

- Năng lượng cung cấp: 433MHz.
- Dải tần số: 420 – 450MHz.
- Năng lượng hoạt động: mặc định 3.3V.
- Chuẩn giao tiếp: SPI.

Bảng 2.6 Các chân của module SX1278

Chân	Tên	Mô tả
1	GND	Ground
2	GND	Chân I/O
3	3.3V	Chân I/O
4	RESET	Power (3.3V)
5	DIO0	Chân I/O
6	DIO1	Chân I/O
7	DIO2	Chân I/O
8	DIO3	Chân I/O
9	GND	Ground
10	DIO4	Chân I/O
11	DIO5	Chân I/O
12	SCK	SPI Clock
13	MISO	SPI dữ liệu ra
14	MOSI	SPI dữ liệu vào
15	NSS	SPI Chip Select
16	GND	Ground

### 2.3.6 Module Sim và usb sim



Hình 2.19 Module 4G SIMCOM A7680C

#### Thông số kỹ thuật:

- Dải năng lượng cung cấp hoạt động rộng từ 4.5 - 16V.
- Mạng sử dụng 4G Cat.1.
- Mức logic I/O TTL( Serial 3.3V).
- Chuẩn giao tiếp: UART.
- Loại ăngten: IPEX 1.
- LTE-FDD hỗ trợ: B1/B3/B5/B8.
- Băng tần hỗ trợ: LTE-TDD: B34/B38/B39/B40/B41.

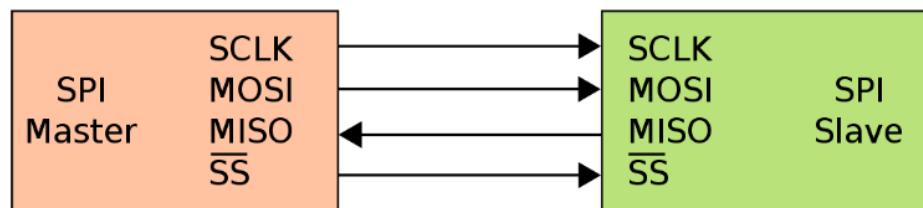
## 2.4 Các chuẩn truyền dữ liệu

### 2.4.1 SPI (Serial Peripheral Interface)

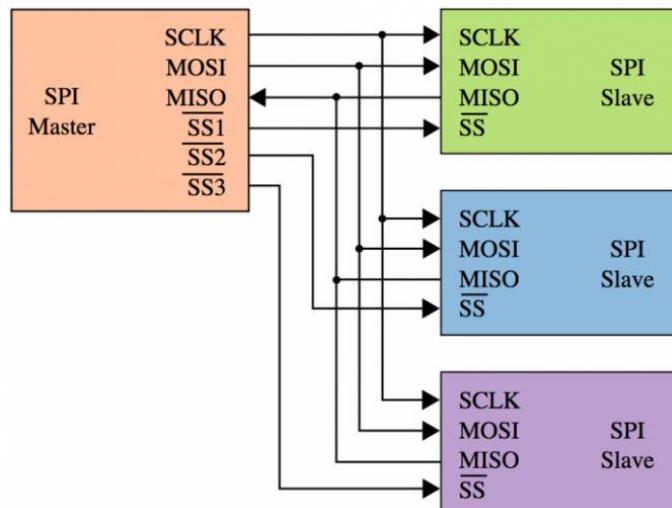
#### 2.4.1.1 Tổng quan

SPI (Serial Peripheral Interface) là một chuẩn giao tiếp đồng bộ ra đời vào năm 1980, được sử dụng để kết nối các thiết bị và có tốc độ truyền tải nhanh hơn so với I2C. Chuẩn này sử dụng 4 dây truyền tín hiệu, với kênh truyền dữ liệu nối tiếp song công (full-duplex), hoạt động theo mô hình Master – Slave.

Bao gồm các đường truyền SCLK tín hiệu cho xung clock từ master, MOSI ngõ ra từ master, MISO vào master và đường tín hiệu SS/CS để lựa chọn thiết bị được thực hiện khi muốn chọn Slave nào thì chỉ cần kéo xuống mức 0 [13].

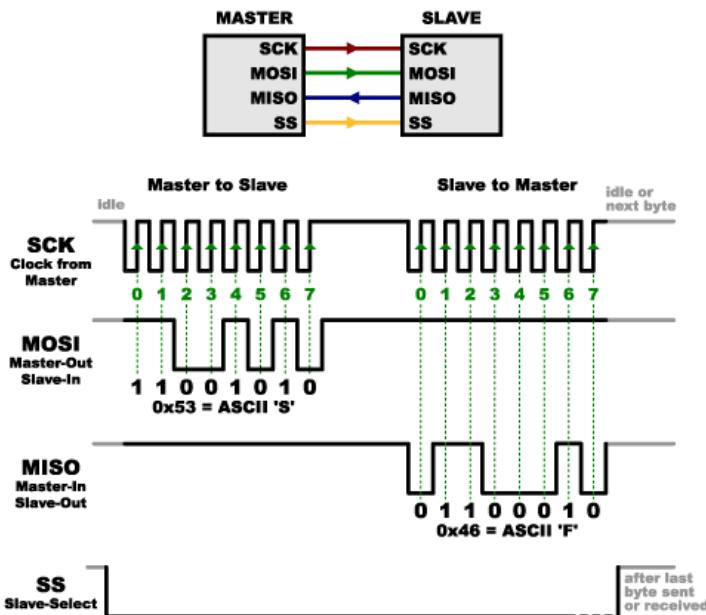


Hình 2.20 Mô hình SPI 1 Master và 1 Slave



Hình 2.21 Mô hình SPI 1 master và nhiều slave

#### 2.4.1.2 Cơ chế hoạt động



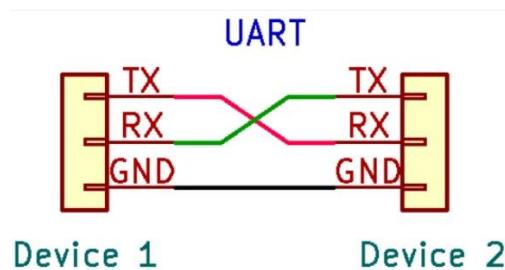
Hình 2.22 Mô tả cơ chế hoạt động SPI

Vì có 1 chân truyền và 1 chân truyền riêng biệt nhau nên có thể truyền và nhận cùng 1 lúc. Trước khi truyền hoặc nhận thì kéo chân SS xuống mức 0 để chọn Slave. Khi Master truyền và nhận đều phải có xung clock. Mỗi lần truyền nhận đều là 8 bits.

#### 2.4.2 UART (*Universal asynchronous receiver transmitter*)

##### 2.4.2.1 Tổng quan

UART là chuẩn giao tiếp truyền nhận dữ liệu nối tiếp không đồng bộ, trong đó đồng hồ (clock) của thiết bị truyền và nhận có thể không khớp với nhau. Giao tiếp này bao gồm hai đường truyền, TX (truyền) và RX (nhận) [13].

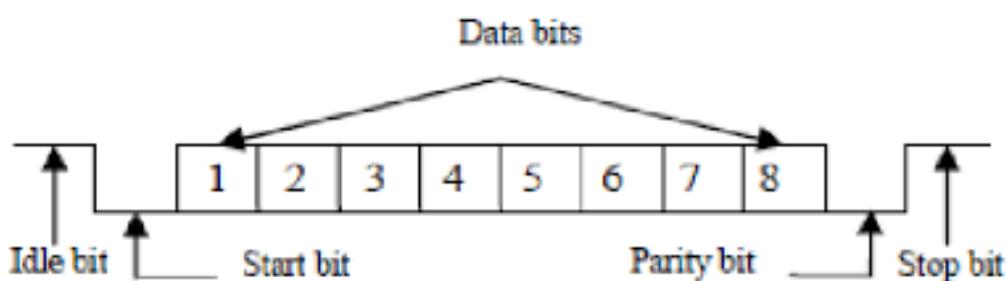


Hình 2.23 Mô hình kết nối Uart

#### 2.4.2.2 Cơ chế hoạt động

Hoạt động theo truyền thông bất đồng bộ. Vì clock hoạt động của bộ Uart của thiết bị truyền và thiết bị nhận có thể khác nhau. Do đó không phân biệt chủ - tớ (master – slave). Các hệ thống là ngang cấp trong chuẩn truyền UART. Do đó khi truyền dữ liệu mỗi hệ thống sử dụng 1 mạch dao động riêng – 2 hệ thống 2 mạch dao động độc lập nhưng phải cùng tần số.

#### 2.4.2.3 Khung truyền



Hình 2.24 Khung truyền Uart

**Idle bit:** trạng thái “none return to zero (NRZ)” trên đường dây. Nó có nghĩa là trong trạng thái nhàn rỗi, tồn tại trạng thái logic-1 tại các đường dây nối tiếp.

**Bit Start:** đường dây điện áp UART được giữ ở mức cao ở trạng thái không tải nên khi bắt đầu truyền, biên độ điện thay đổi chuyển đổi trạng thái cao sang trạng thái thấp và trở lại trạng thái cao trong chu kỳ. Đó chính là dấu hiệu để người nhận chuẩn bị sẵn sàng cho dữ liệu đến.

**Data:** Sau khi dữ liệu bit bắt đầu được truyền đi. Khung dữ liệu chứa dữ liệu thực tế, hữu ích. Kích thước của frame từ 5-9 bit. Còn có tính chẵn lẻ, kích thước tối đa là 8 bit và nếu không sử dụng tính chẵn lẻ, kích thước tối đa có thể là 9 bit. Những bit dữ liệu này gửi với bit ít quan trọng nhất trước tiên.

**Parity bit:** Sau khung dữ liệu, bit chẵn lẻ có thể được đặt - nó là tùy chọn. Nó được sử dụng để phát hiện lỗi, để xem liệu dữ liệu có bị thay đổi hay không. Giá trị của bit chẵn lẻ phụ thuộc vào loại chẵn lẻ được sử dụng: chẵn hay lẻ. Nếu chúng ta sử dụng

chẵn lẻ thì số đơn vị (1) phải là số chẵn. Trong chẵn lẻ lẻ, số lượng đơn vị phải là số lẻ. Ví dụ: nếu chúng ta có dữ liệu 11001010, số lượng đơn vị là 4, là số chẵn. Nếu chúng ta sử dụng chẵn lẻ, bit chẵn lẻ sẽ là 0. Nhưng nếu chúng ta sử dụng chẵn lẻ lẻ, bit chẵn lẻ sẽ là 1.

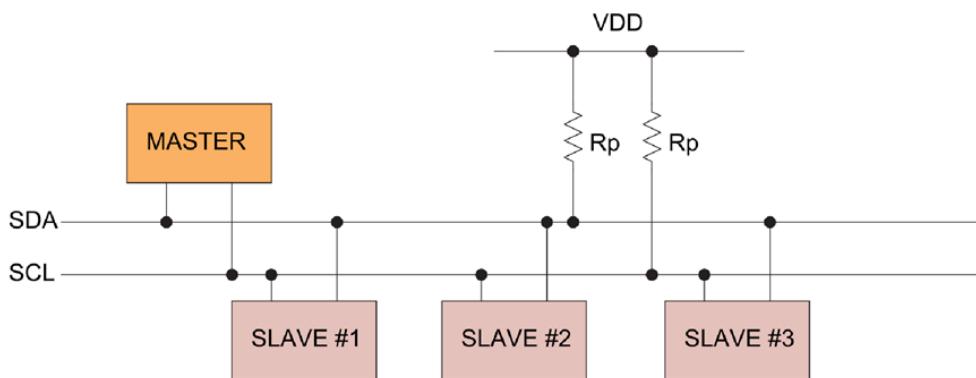
Khi sử dụng một bit chẵn lẻ, bit chẵn lẻ chỉ có thể phát hiện một bit bị đảo lộn. Nếu số bit chẵn bị lật thì không có cách nào để phát hiện điều đó một cách đáng tin cậy.

**Stop bit:** Khi muốn kết thúc gói, bit dừng được sử dụng. Ngược lại với bit start, mức điện áp trên đường truyền đi từ mức thấp đến mức cao đối với một hoặc hai bit. Bit dừng thứ hai là tùy chọn và nó thông thường được dùng để chuyển giao cho người nhận đủ thời gian để sẵn sàng cho khung hình tiếp theo, nhưng đây là thông lệ không phổ biến.

### 2.4.3 I2C (Inter-Integrated Circuit)

#### 2.4.3.1 Tổng quan

I2C được phát triển bởi Philips Semiconductors. I2C bao gồm 2 dây là SCL và SDA. Một chuẩn giao tiếp tại một thời điểm chỉ một master, truyền dữ liệu SDA (serial data) và truyền tải xung SCL (serial clock) [13].



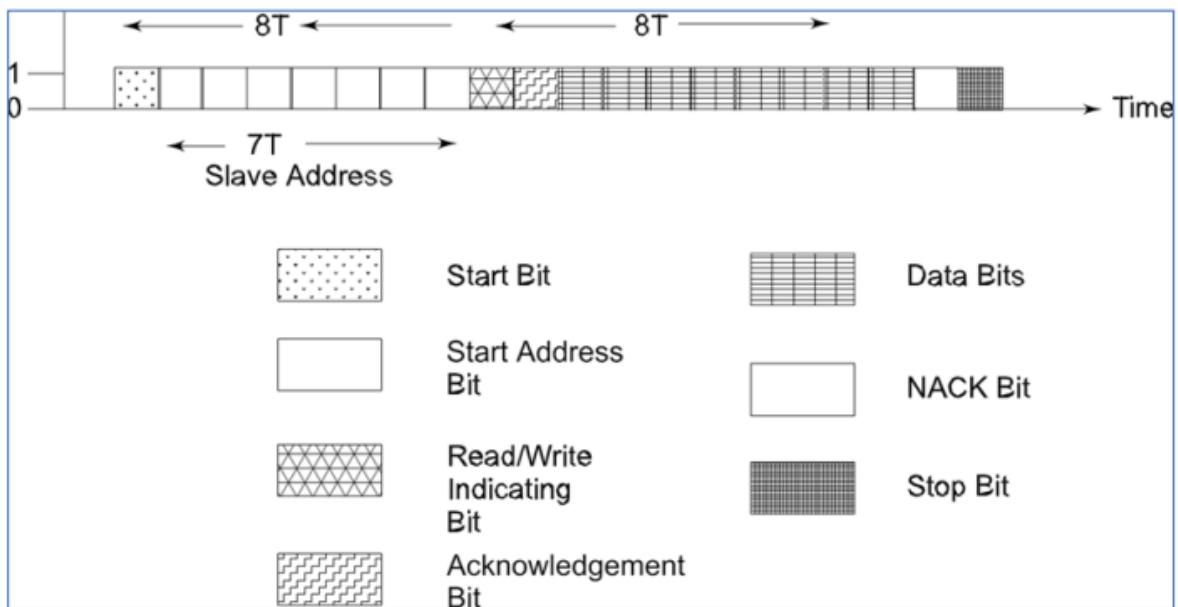
Hình 2.25 Mô hình kết nối I2C

#### *2.4.3.2 Cơ chế hoạt động*

Quá trình giao tiếp sử dụng các lệnh gửi và nhận dữ liệu bằng giao thức I2C, bao gồm:

- Lệnh gửi START: master gửi tín hiệu START trên dây SDA để bắt đầu phiên giao tiếp.
- Gửi địa chỉ của slave: master gửi địa chỉ của slave mà nó muốn truyền hoặc nhận dữ liệu đến tất cả các slave kết nối.
- Slave phản hồi: slave tương ứng với địa chỉ được gửi sẽ phản hồi ACK (acknowledge) bằng cách đặt mức logic 0 lên dây SDA.
- Truyền/nhận dữ liệu: sau khi slave đã phản hồi ACK, master có thể tiếp tục gửi hoặc nhận dữ liệu từ slave. Khi truyền dữ liệu, master sẽ gửi dữ liệu trên dây SDA và slave phản hồi ACK sau mỗi byte dữ liệu. Khi nhận dữ liệu, master sẽ đặt dây SDA vào trạng thái đọc và slave sẽ gửi dữ liệu trên dây SDA và master sẽ phản hồi ACK sau mỗi byte dữ liệu.
- Lệnh gửi STOP: khi master đã truyền/nhận đủ dữ liệu, nó sẽ gửi lệnh STOP trên dây SDA để kết thúc phiên giao tiếp.

### 2.4.3.3 Khung truyền



Hình 2.26 Khung truyền I2C

**Start bit:** Báo hiệu một gói dữ liệu sắp truyền tới.

**Start Address bit:** Xác định địa chỉ Slave, mà đang được gửi các khung dữ liệu (gồm nhiều bytes) của master.

**R/W bit:** Xác định một chu kỳ là đọc hoặc ghi trong tiến trình.

**ACK bit:** Xác định các dữ liệu hiện nay là một ACK (từ slave).

**Data bit:** I2C byte dữ liệu.

**NACK bit:** NACK (Negative ACK) từ phía nhận. Nếu sau khi đã gửi NACK, sau đó ACK, sau đó ACK sau khi chuyển dữ liệu thành công là không cần thiết từ các slave, ngược lại ACK cần được gửi từ slave.

**Stop bit:** Báo hiệu đã truyền xong.

## CHƯƠNG 3 PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

### 3.1 Phân tích hệ thống

#### 3.1.1 Yêu cầu hệ thống

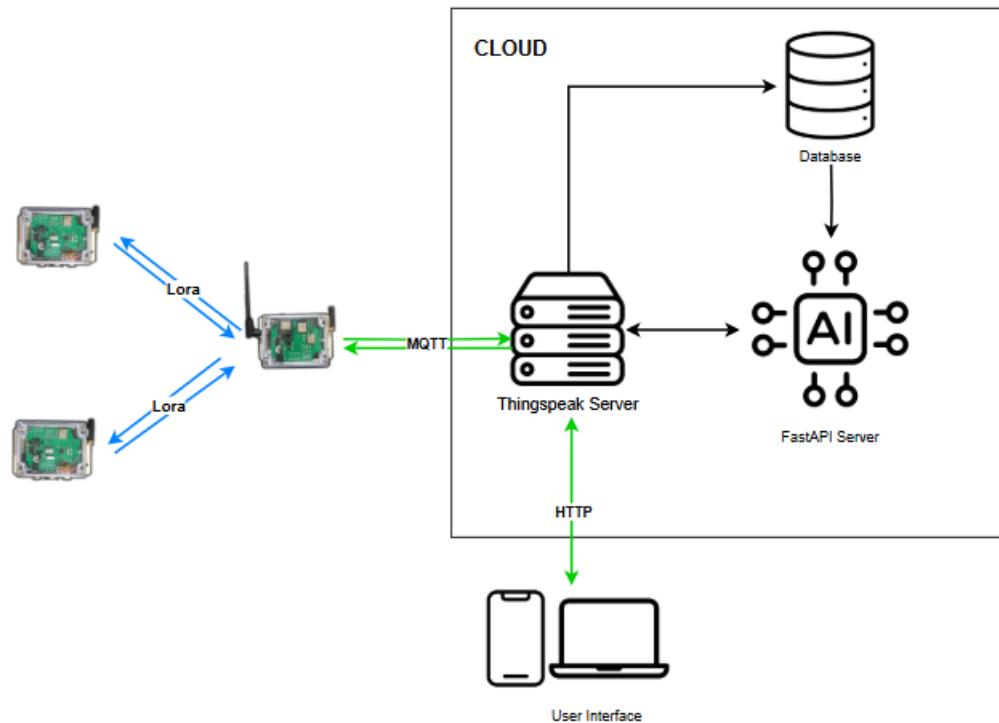
##### - Phần cứng:

- Board mạch sử dụng thu thập dữ liệu từ cảm biến đo độ mặn, nhiệt độ, độ ẩm,...
- Có các chức năng giao tiếp không dây Lora để kết nối các trạm đo gần đó.
- Có module sim 4G để gửi dữ liệu lên server.

##### - Phần mềm:

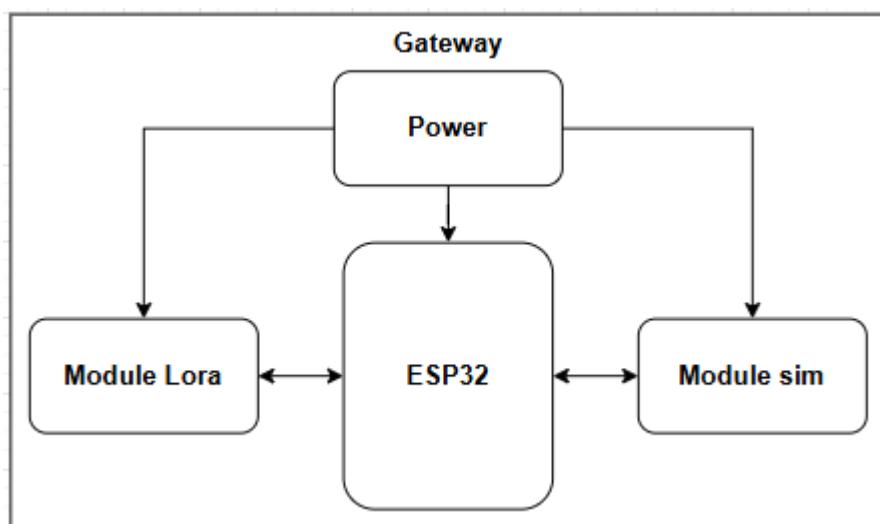
- Chương trình thu thập dữ liệu gửi lên server.
- Có khả năng kết nối các trạm gần đó tạo thành mạng để chuyển tiếp/gửi dữ liệu.
- Chương trình giám sát hoạt động các trạm đo.
- Phần mềm phải có thêm các tiêu chuẩn bảo mật dữ liệu.

### 3.1.2 Sơ đồ khái niệm



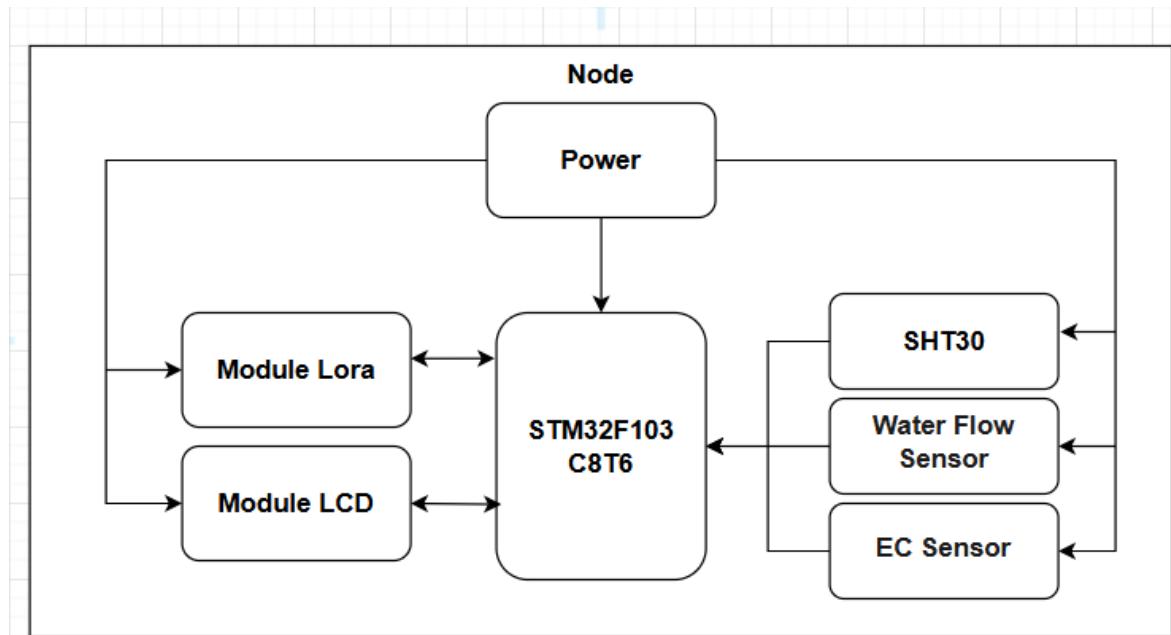
Hình 3.1 Sơ đồ khái niệm

Với yêu cầu đi lắp đặt thực tế, nên nhóm đã quyết định chọn server public là thingspeak để thuận tiện cho việc xem được dữ liệu thông qua app.



Hình 3.2 Sơ đồ khái niệm Gateway

Với gateway việc chọn ESP32 nhằm mục đích có thể linh hoạt hơn khi sử dụng 4G hoặc wifi.



Hình 3.3 Sơ đồ khái niệm Node

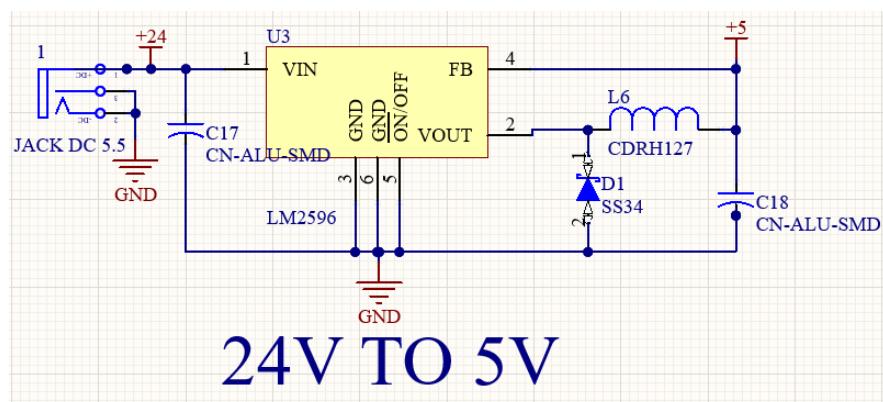
Node nhóm quyết định dùng STM32F103C8T6 phù hợp để đọc cảm biến và đồng thời gửi dữ liệu về gateway qua LoRa.

### 3.2 Thiết kế hệ thống

#### 3.2.1 Phản ứng

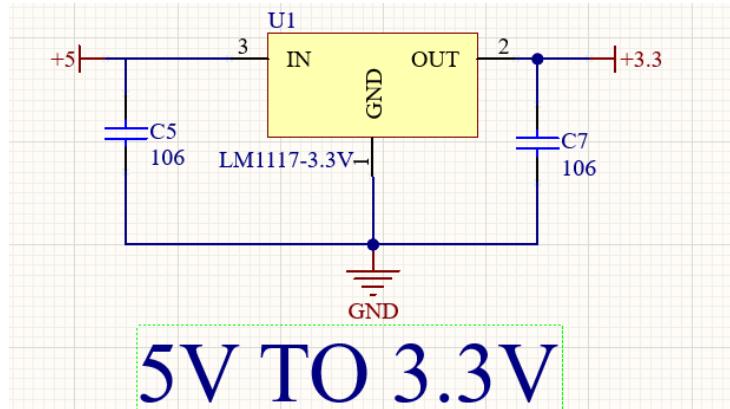
##### 3.2.1.1 Node

- Khối nguồn



Hình 3.4 Schematic nguồn cho các cảm biến

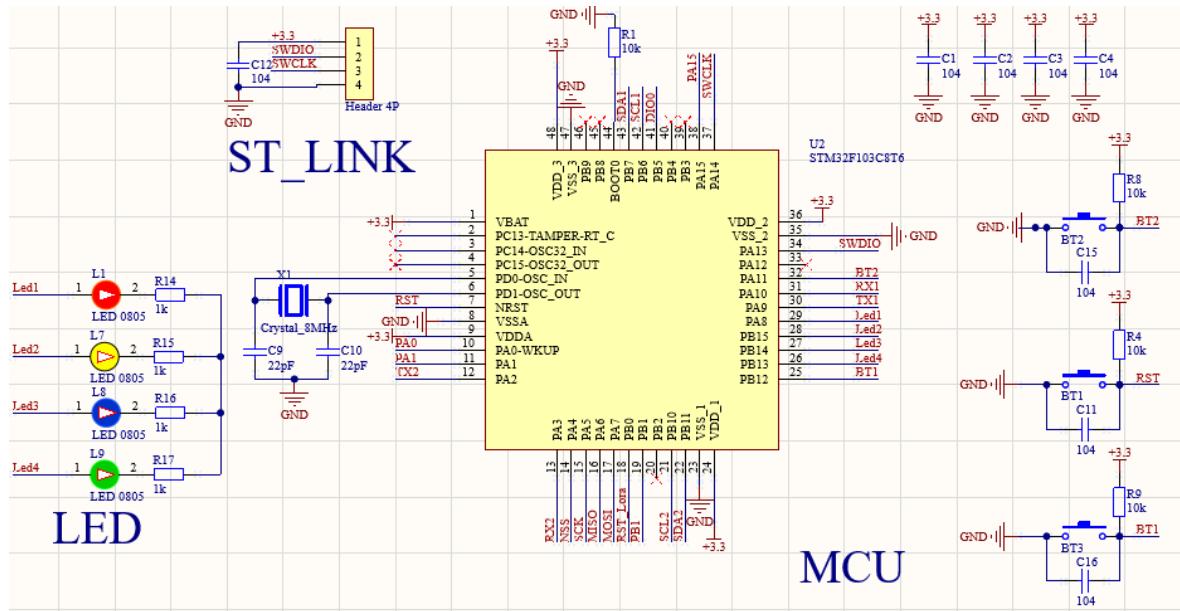
Mạch Node sử dụng các loại cảm biến EC, SHT30, cảm biến lưu lượng đều hoạt động ở mức điện áp 5V. Vì vậy nhóm sử dụng IC LM2596 để hạ áp từ 24V xuống 5V. Chọn điện áp đầu vào là 24V vì cho cảm biến mở rộng trong tương lai cho dạng cảm biến công nghiệp.



Hình 3.5 Schematic nguồn cho STM32F103C8T6

Mạch node sử dụng MCU là STM32F103C8T6, sử dụng điện áp 3.3V nên nhóm chọn IC LM1117 để hạ áp từ 5V xuống 3.3V cho MCU.

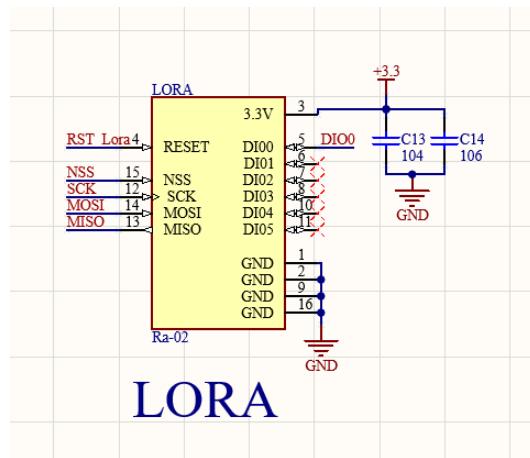
- Khối MCU



Hình 3.6 Schematic STM32F103C8T6 ra chân

Với STM32F103C8T6 nhóm sử dụng thạch anh 8MHz và đồng thời kết nối với 2 tụ 22pF để tạo ổn định tần số dao động của thạch anh. Để nạp code cho MCU thì sử dụng stlink. Ngoài ra nhóm còn sử dụng 4 tụ 104 (100nF) để lọc nhiễu đảm bảo rằng mcu sẽ hoạt động ổn định.

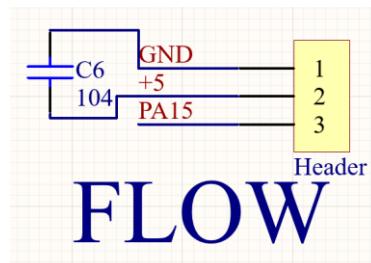
- Khối LoRa



Hình 3.7 Schematic LoRa trên board Node

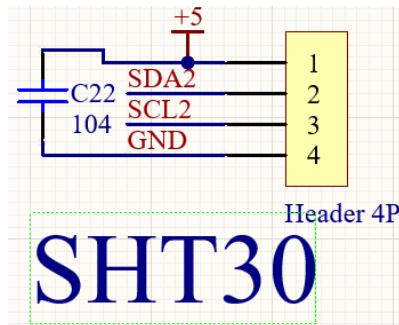
Node sử dụng module LoRa Ra-02 AI-Thinker và trong thiết kế thì sử dụng thêm 2 tụ 104 (100nF) để lọc nguồn tạo sự hoạt động ổn định.

- Khối cảm biến



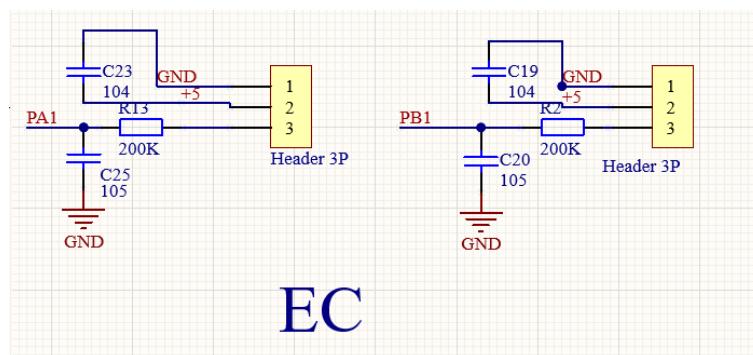
Hình 3.8 Schematic ra chân kết nối cảm biến lưu lượng

Cảm biến lưu lượng sử dụng chân GPIO để đọc xung ngắn từ cảm biến. Ngoài ra còn có tụ lọc nguồn cho cảm biến hoạt động ổn định hơn.



Hình 3.9 Schematic ra chân kết nối cảm biến SHT30

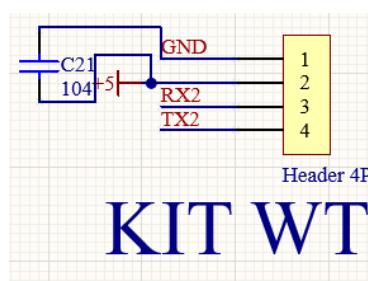
Cảm biến SHT30 sử dụng chuẩn giao tiếp I2C, hoạt động mức điện áp 5V và có tụ lọc nguồn cấp để tạo sự ổn định cho cảm biến.



Hình 3.10 Schematic ra chân kết nối cảm biến độ mặn

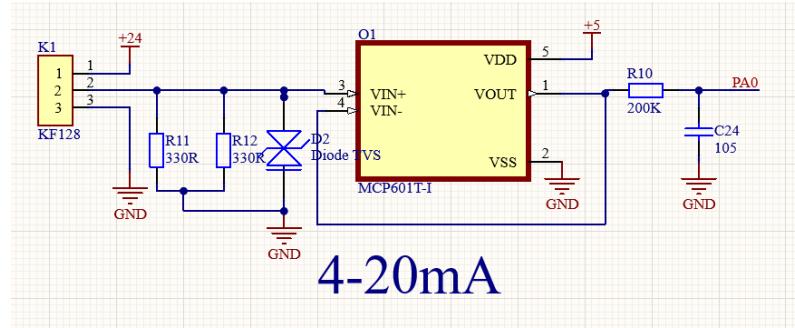
Cảm biến độ mặn sử dụng ADC để đọc tín hiệu gửi về từ cảm biến có bộ lọc thông thấp RC dưới 1Hz. Còn có tụ lọc 104 (100nF) để lọc nhiễu tạo ổn định cho cảm biến.

- Khối cảm biến mở rộng



Hình 3.11 Schematic đọc kit thời tiết

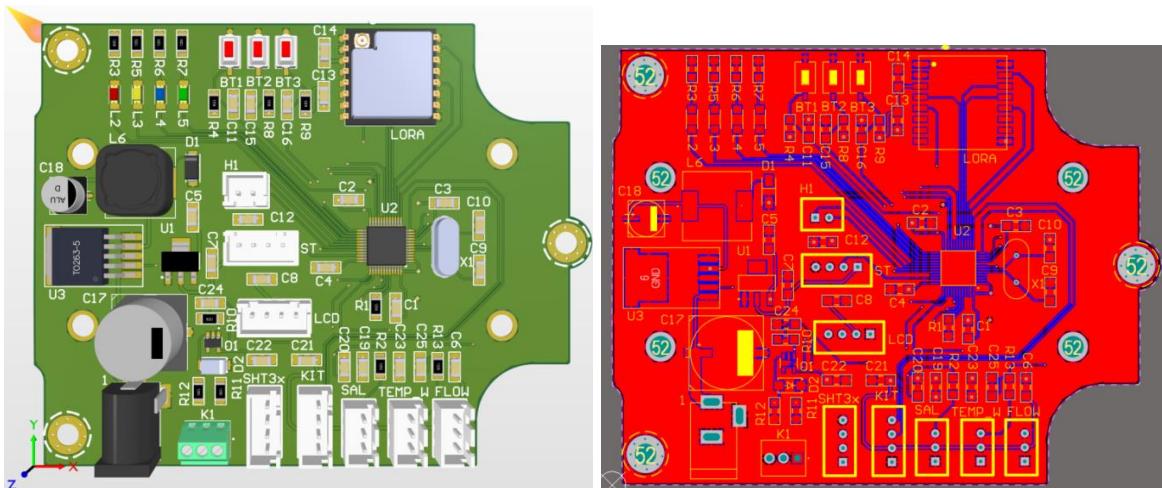
Tương lai sẽ mở rộng thêm để đọc từ kit thời tiết lấy thông số gió, lượng mưa.



Hình 3.12 Schematic đọc cảm biến 4-20mA

Có còn thêm cảm biến 4-20mA để mở rộng trong tương lai.

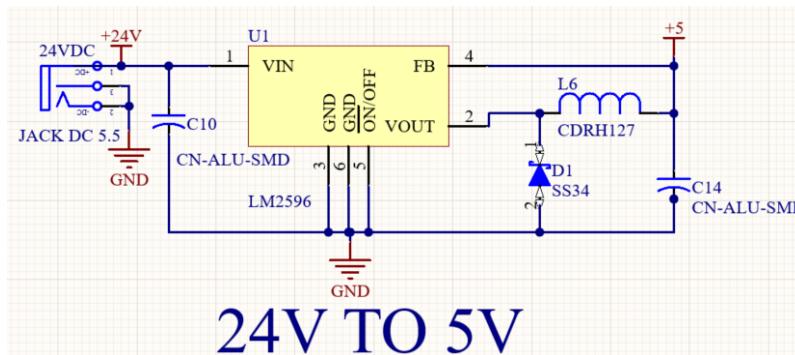
Sau khi hoàn thành, chuyển sang thiết kế pcb để in mạch node.



Hình 3.13 Thiết kế layout cho mạch node

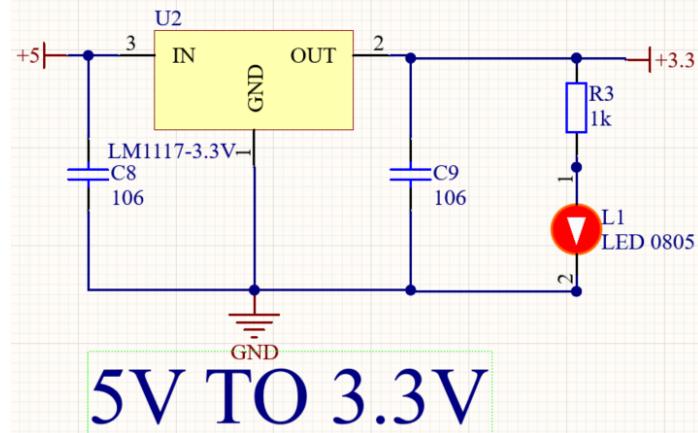
### 3.2.1.2 Gateway

- Khối nguồn



Hình 3.14 Schematic nguồn cho module sim

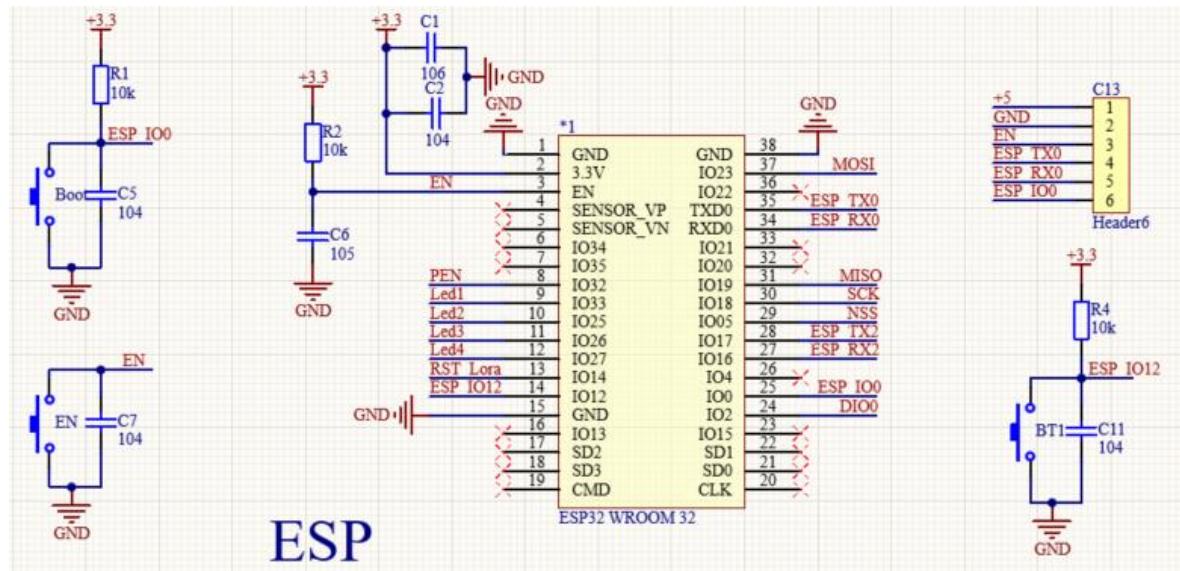
Mạch Gateway có module sim hoạt động ở mức điện áp 5V. Vì vậy nhóm sử dụng IC LM2596 để hạ áp từ 24V xuống 5V.



Hình 3.15 Schematic nguồn cho module sim

Mạch Gateway dùng ESP32 và module LoRa hoạt động ở mức điện áp 3.3V. Nên nhóm chọn IC LM1117 để hạ áp từ 5V xuống 3.3V.

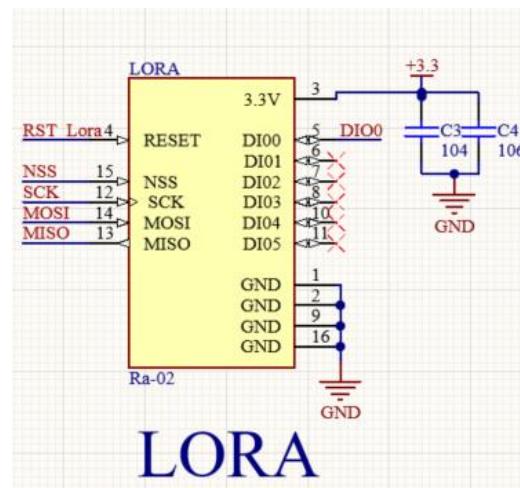
- Khối ESP



Hình 3.16 Schematic ESP32

Để linh hoạt về wifi và 4G nhóm sử dụng ESP32 vì có thể sử dụng kết nối wifi, cũng có thể gửi dữ liệu lên server thông qua module sim.

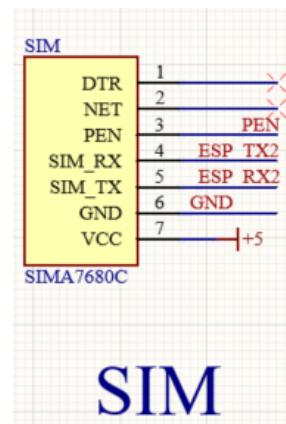
- Khởi LoRa



Hình 3.17 Schematic LoRa trên board gateway

Gateway sử dụng module LoRa Ra-02 AI-Thinker và trong thiết kế thì sử dụng thêm 2 tụ 104 (100nF) để lọc nguồn tạo sự hoạt động ổn định.

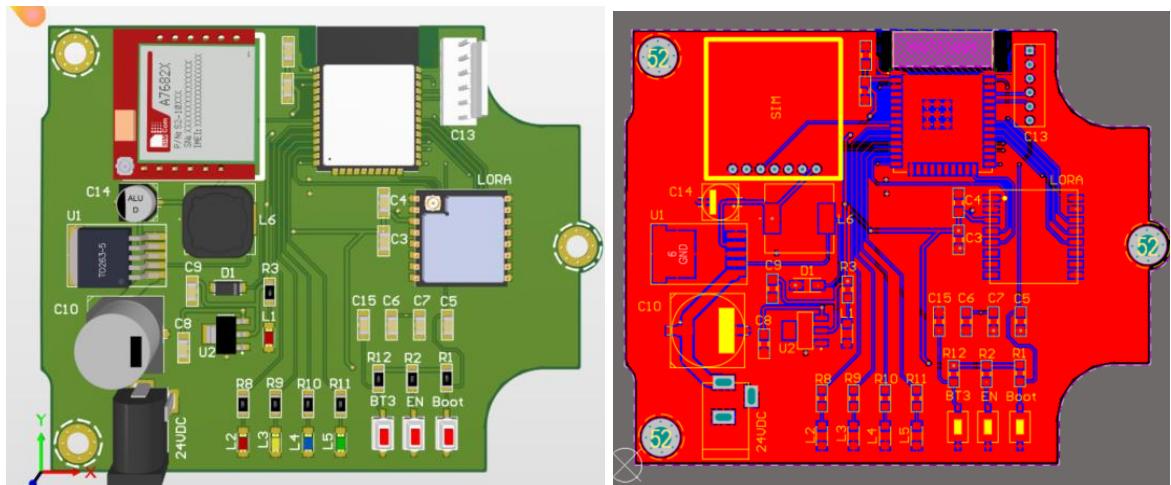
- Khởi module sim



Hình 3.18 Schematic chân kết nối module sim

Gateway sử dụng module sim để dự phòng ngừa trong trường hợp không có wifi, để gửi dữ liệu lên server.

Thiết kế layout, cho gateway để in mạch.

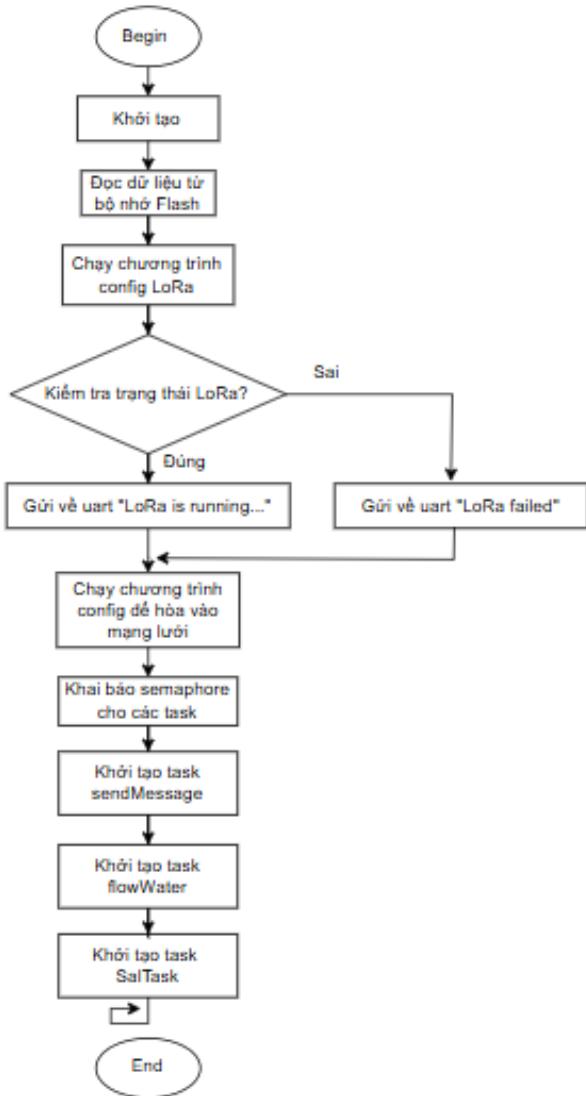


Hình 3.19 Thiết kế layout cho mạch gateway

### 3.2.2 Phần mềm

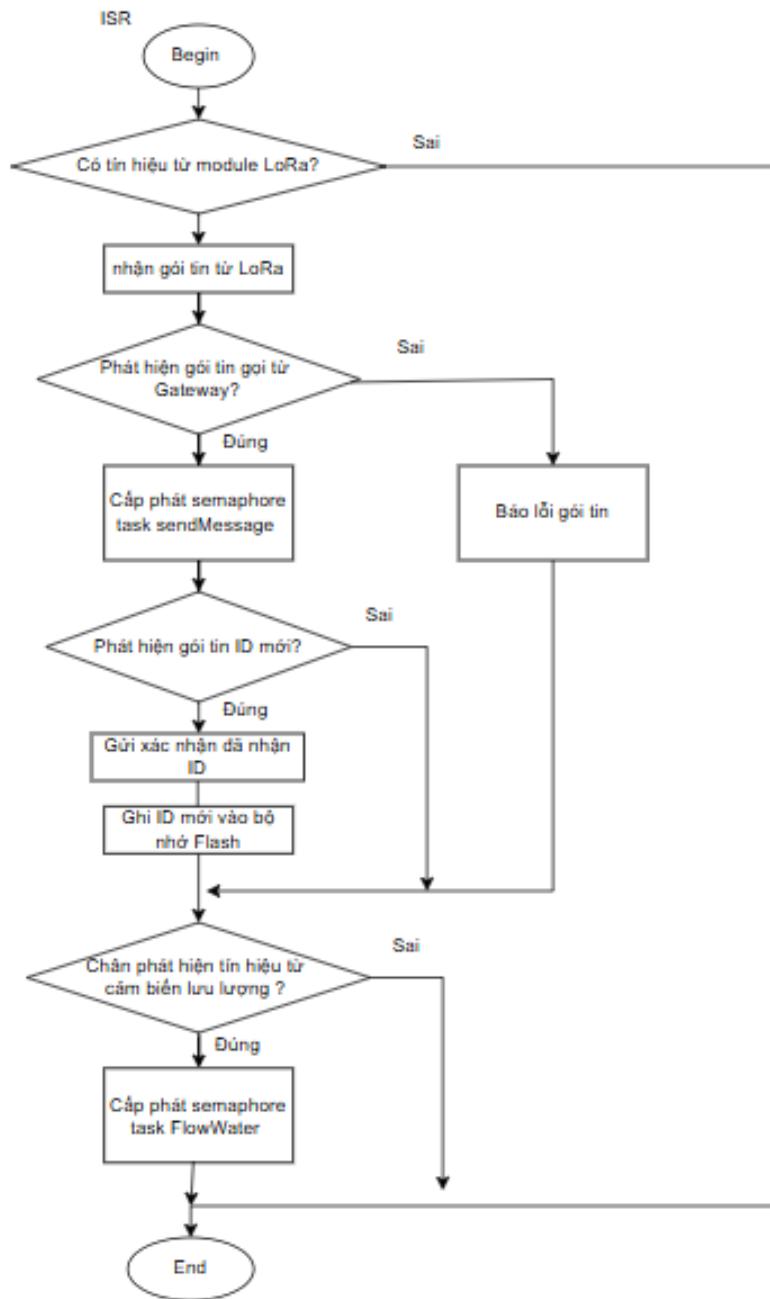
#### 3.2.2.1 Node

Mạch node khi bật lên các thì đầu tiên sẽ khởi tạo, đọc ID của chính nó và ID các node cận nếu có. Sẽ có chương trình config để hòa nhập vào mạng lưới LoRa, đồng thời node chịu trách nhiệm đọc các giá trị cảm biến SHT30, cảm biến lưu lượng và cảm biến độ mặn. Trường hợp lúc test có thêm cấu hình để điều khiển LCD 16x2. Chương trình trên node chia ra thành 3 task, task gửi giữ liệu nếu có gateway gọi, task đọc cảm biến EC, task đọc cảm biến lưu lượng.



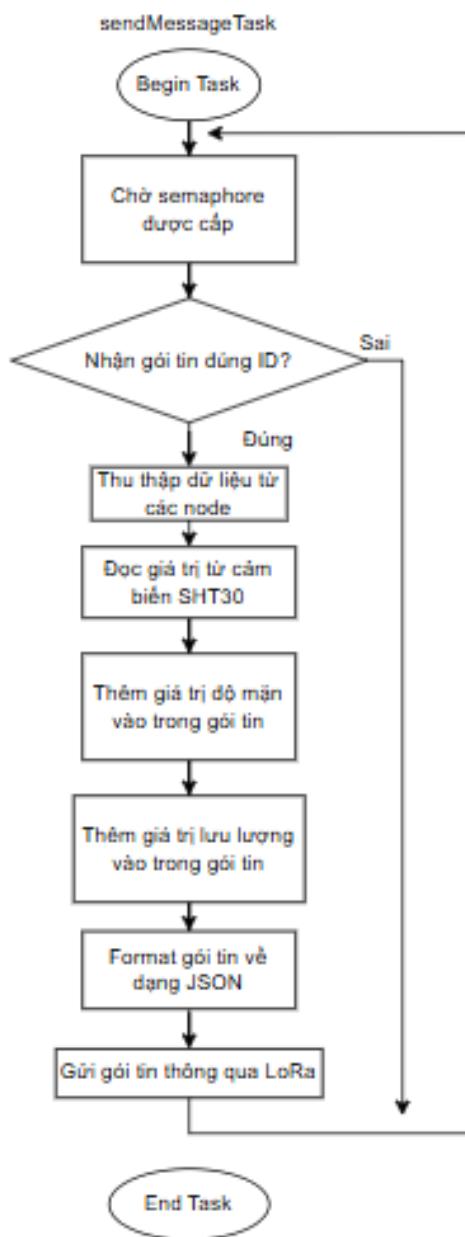
Hình 3.20 Lưu đồ chương trình chính Node

Chương trình ngắn trên node, dùng để phát hiện tín hiệu từ LoRa để nhận gói tin kịp thời. Đồng thời đọc gói tin này để biết nhiệm vụ gì để cấp semaphore cho các task khác hoạt động, cũng phát hiện các node lạ nếu đúng ID thì cho hòa vào mạng lưới.

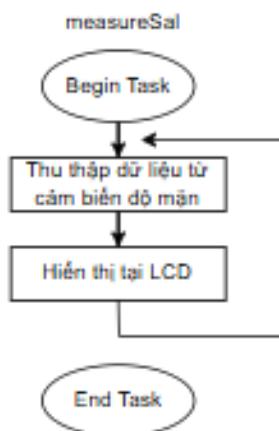


Hình 3.21 Lưu đồ chương trình ngắt điều khiển các task

Ở task gửi dữ liệu, khi trong ngắt nhận được gói tin là “ID” thì được cấp semaphore để task gửi tin này hoạt động. Trong task này sẽ đọc giá trị cảm biến SHT30 và sẽ đọc lọc qua bộ lọc trung vị để lấy giá trị. Sau đó dữ liệu được đóng gói về dạng JSON gửi lại cho gateway.

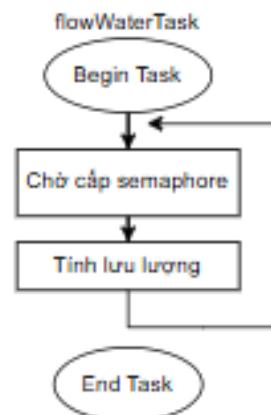


Hình 3.22 Lưu đồ chương trình task gửi dữ liệu



Hình 3.23 Lưu đồ chương trình task đọc dữ liệu từ cảm biến độ mặn

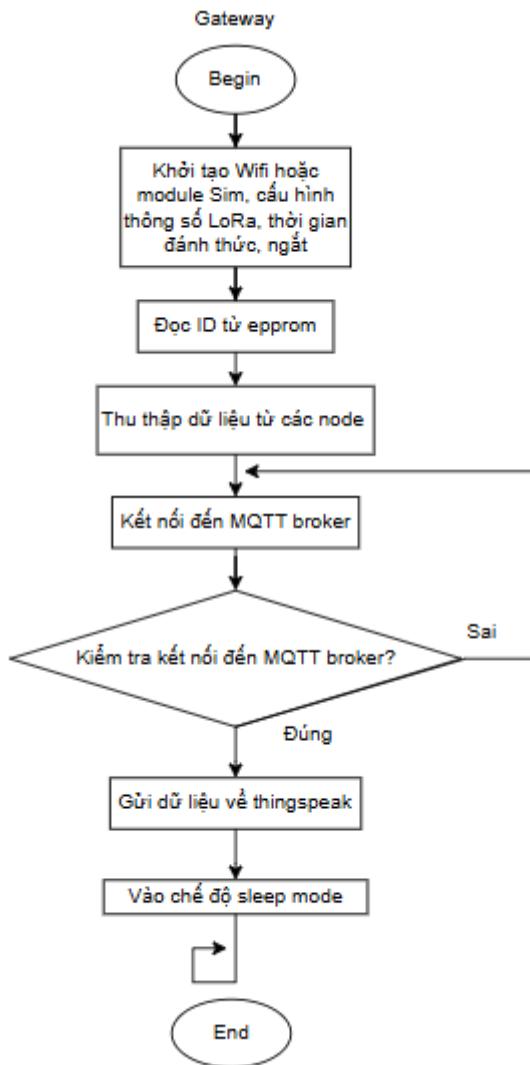
Task “measureSal” để đọc cảm biến độ dẫn điện trong nước sau đó được qua bộ lọc trung vị để lấy giá trị. Giá trị độ mặn nhôm đã dùng công thức tính từ giá trị độ dẫn điện đọc được, độ mặn (ppt) = EC (mS/cm) \* 0.64 [14].



Hình 3.24 Lưu đồ chương trình task đọc lưu lượng nước

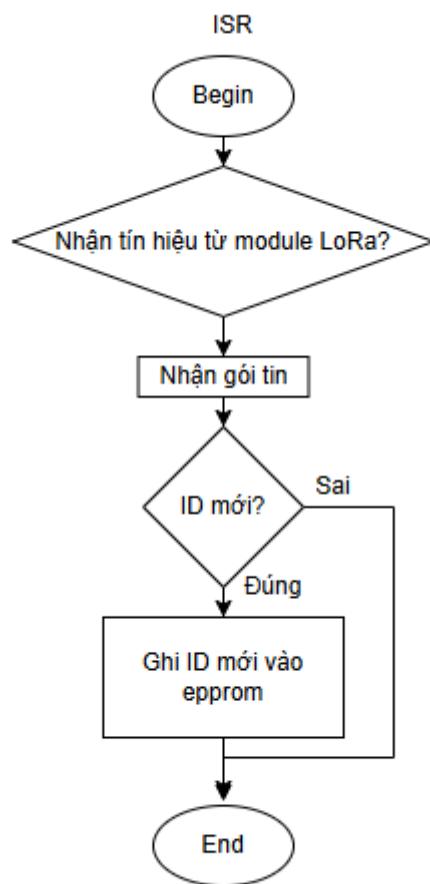
Task này hoạt động dựa vào ngắt ngắt phát hiện xung từ cảm biến lưu lượng, sau đó cấp semaphore để task này đếm và tính lưu lượng cập nhập giá trị.

### 3.2.2.2 Gateway



Hình 3.25 Lưu đồ chương trình chính gateway

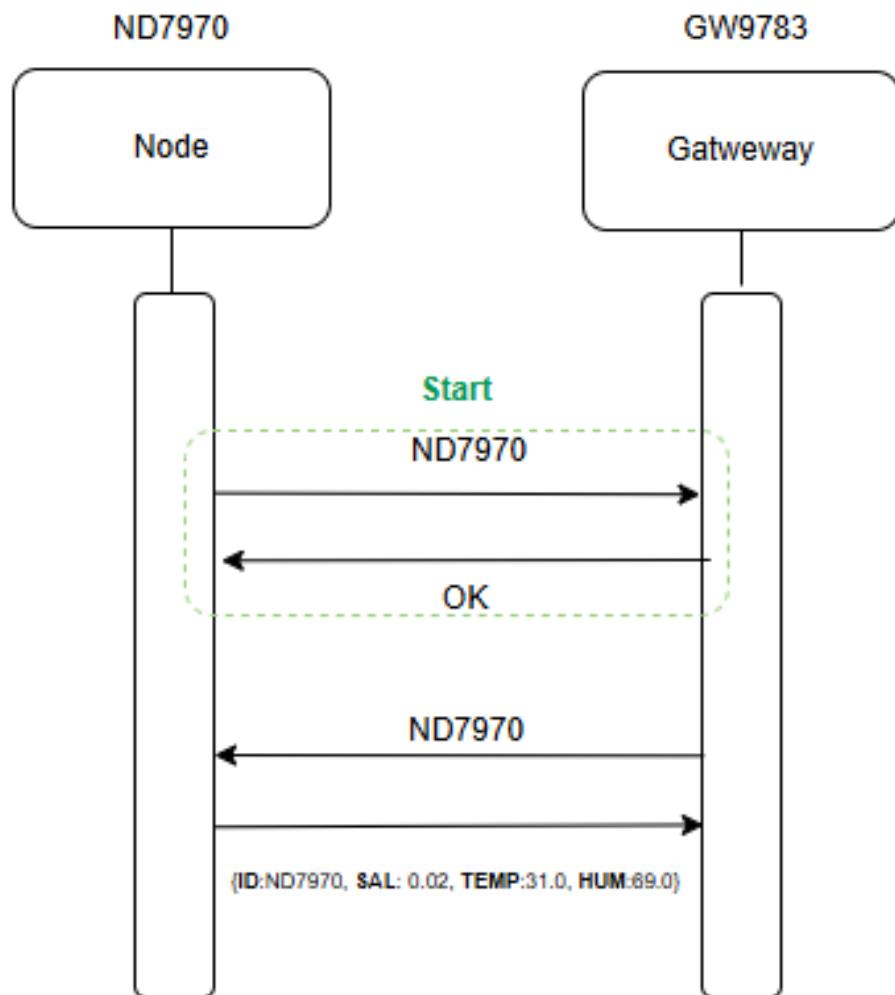
Khi bắt đầu gateway cấu hình wifi hoặc module sim, cấu hình LoRa. Để tiết kiệm năng lượng thì gateway sẽ có vào chế độ “deep sleep”, thì ban đầu gateway sẽ cấu hình thông số để thức dậy. Các ID của chính nó và các node được lưu vào eeprom. Sau đó, dùng những ID của node để gọi đến nó để kêu gọi node gửi dữ liệu lại cho gateway. Gọi hết ID để thu dữ liệu, đóng gói theo cấu trúc của thingspeak để gửi dữ liệu được lên. Hoàn thành hết tất cả việc thì gateway tắt module sim, vào chế độ “deep sleep”.



Hình 3.26 Lưu đồ chương trình ngắt gateway

Chương trình ngắt trên gateway, đóng vai trò nhận ID của node mới và ghi vào eeprom.

### 3.2.2.3 Cấu trúc dữ liệu



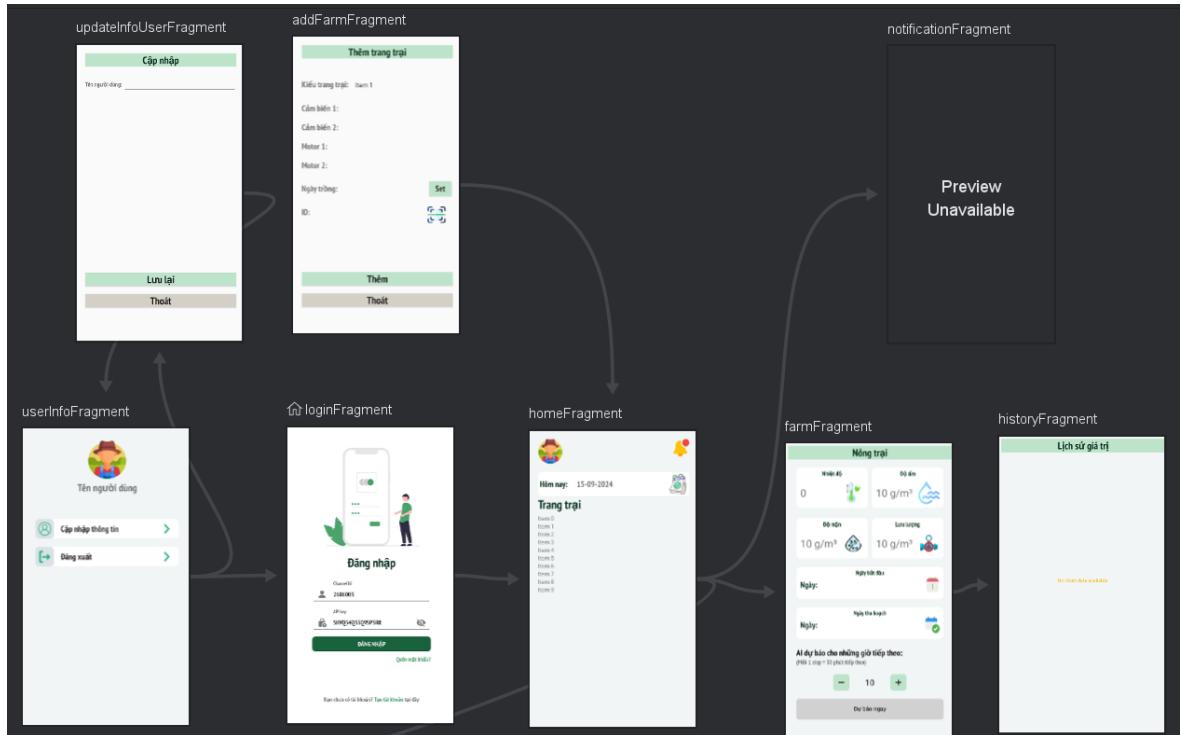
Hình 3.27 Giao tiếp giữa node và Gateway

Cấu trúc dữ liệu ở đây nhóm chọn là JSON. Có dạng:

“{ID: ND7970, SAL:0.02, TEMP: 31.0, HUM:69.0}”

- ID là mã id đã được định danh cho node.
- SAL là thông số của độ mặn.
- TEMP là thông số nhiệt độ.
- HUM là thông số độ ẩm.

### 3.2.2.4 App người dùng



Hình 3.28 Các luồng hoạt động của app

Ở trong app sử dụng này, nhóm các layout như phần login để đăng nhập bằng ID và mật mã người dùng. Sau khi đăng nhập vào sẽ có các nông trại cùng với tên được hiện ra ứng với tài khoản, đồng thời có icon chuông thông báo và icon ảnh đại diện, cùng với thông tin ngày hiện tại. Khi nhấn vào icon ảnh giao diện, có thể thay đổi thông tin người dùng và nhấn lưu. Chuông thông báo xem được cảnh báo người dùng về độ mặn vượt mức.

Nhấn vào từng nông trại, sẽ xem được thông tin cảm biến lấy được từ hệ thống các node và gateway. Trong đó, nhấn vào từng giá trị hiện tại đó xem hiển thị ra biểu đồ lịch sử giá trị. Phần phát triển thêm vào đó là thành phần dự đoán các thông số trong tương lai, người dùng có thể chọn vào số điểm muốn dự đoán mỗi điểm 30 phút, sau đó chỉ cần nhấn dự đoán sẽ đưa ra biểu đồ các giá trị dự đoán.

### **3.2.3 Xây dựng mô hình AI dự đoán các giá trị trong tương lai**

#### **3.2.3.1 Dữ liệu chuỗi thời gian**

Dữ liệu chuỗi thời gian là một tập hợp các điểm dữ liệu được thu thập, ghi lại hoặc quan sát theo thứ tự thời gian. Dữ liệu này thường được ghi lại ở các khoảng thời gian cách nhau đều đặn (ví dụ: hàng giây, hàng phút, hàng giờ, hàng ngày) hoặc không đều đặn nhưng luôn có thứ tự thời gian rõ ràng.

Đặc điểm chính của dữ liệu chuỗi thời gian:

- Trình tự thời gian: Các điểm dữ liệu được ghi lại theo một trình tự thời gian nhất định, và thứ tự này rất quan trọng để phân tích.
- Tính thời gian: Thời gian là một yếu tố quan trọng trong dữ liệu, và giá trị tại một thời điểm có thể phụ thuộc vào giá trị tại các thời điểm trước đó (tính tự tương quan).
- Xu hướng (Trend): Dữ liệu có thể có xu hướng tăng hoặc giảm theo thời gian.
- Tính chu kỳ (Seasonality): Dữ liệu có thể có các mẫu lặp lại theo thời gian, ví dụ như doanh số bán hàng thường tăng vào cuối năm.
- Nhiễu (Noise): Dữ liệu thường chứa nhiễu, là các biến động không thể dự đoán hoặc không có quy luật.

### 3.2.3.2 Chuẩn bị và xử lý dữ liệu

Dữ liệu được lấy từ thingspeak thông qua API, sau đó sẽ tiến hành phân tích và xử lí.

```
9  def prepare_data():
10 # Load file dữ liệu csv đã lưu sẵn
11 csv_path = os.path.join(DATA_DIR, FILENAME)
12 df = pd.read_csv(csv_path)
13
14 print("trước khi xử lí")
15 print(df.info())
16 print(df.describe())
17 date_time = pd.to_datetime(df.pop('created_at'))
18
19 # giữ lại các cột cần thiết
20 columns_to_keep = ['field1', 'field2', 'field3']
21 df_filtered = df[columns_to_keep]
22
23 print("Sau khi loại bỏ các cột không cần thiết")
24 print(df_filtered.info())
25 print(df_filtered.describe())
26
27 # Đổi tên các cột
28 df = df.rename(columns={"field1": "Temp", "field2": "Humi", "field3": "Sal"})
29 plot_cols = ['Temp', 'Humi', 'Sal']
30 plot_features = df[plot_cols]
31 plot_features.index = date_time
32 _ = plot_features.plot(subplots=True)
33 plt.show()
```

Hình 3.29 Lấy dữ liệu từ Thingspeak và tiến hành phân tích

Dữ liệu trước khi xử lí:

```
Rangetype: 16896 entries, 0 to 16895
Data columns (total 13 columns):
 #   Column      Non-Null Count  Dtype  
 ---  --          --          --    
 0   created_at  16896 non-null   object 
 1   entry_id    16896 non-null   int64  
 2   field1      16896 non-null   float64
 3   field2      16894 non-null   float64
 4   field3      16894 non-null   float64
 5   field4      0  non-null     float64
 6   field5      0  non-null     float64
 7   field6      0  non-null     float64
 8   field7      0  non-null     float64
 9   latitude    0  non-null     float64
 10  longitude   0  non-null     float64
 11  elevation   0  non-null     float64
 12  status      16892 non-null   object 
dtypes: float64(10), int64(1), object(2)
memory usage: 1.7+ MB
None
      entry_id    field1      field2      field3      field4      field5      field6      field7      latitude      longitude      elevation
count  16896.000000  16896.000000  16894.000000  16894.000000  0.0        0.0        0.0        0.0        0.0        0.0        0.0
mean   11882.500000  28.194081   78.369471   0.055672   NaN        NaN        NaN        NaN        NaN        NaN        NaN
std    4877.599410   1.955180   7.882437   0.026399   NaN        NaN        NaN        NaN        NaN        NaN        NaN
min    3435.000000   0.000000   51.510000   0.000000   NaN        NaN        NaN        NaN        NaN        NaN        NaN
25%   7658.750000   26.780000   73.792500   0.040000   NaN        NaN        NaN        NaN        NaN        NaN        NaN
50%   11882.500000   27.930000   79.750000   0.060000   NaN        NaN        NaN        NaN        NaN        NaN        NaN
75%   16106.250000   29.500000   84.047500   0.070000   NaN        NaN        NaN        NaN        NaN        NaN        NaN
max    20330.000000   33.510000   94.290000   0.800000   NaN        NaN        NaN        NaN        NaN        NaN        NaN
```

Hình 3.30 Phân tích dữ liệu

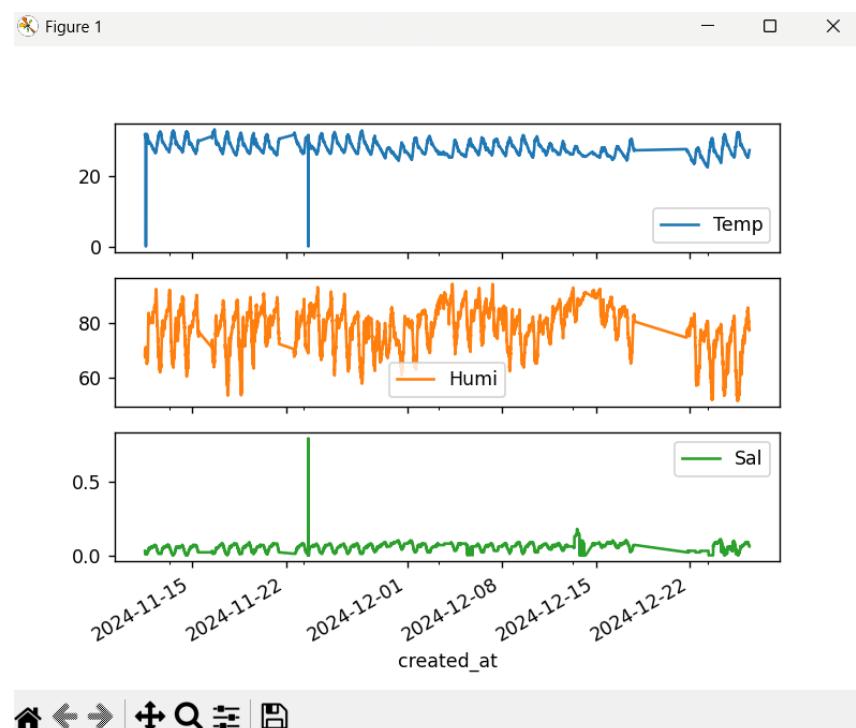
Bộ dữ liệu có nhiều đặc trưng không sử dụng cho mô hình vì vậy cần xóa các cột này. Có tổng cộng 16896 mẫu dữ liệu và không có mẫu nào bị null. Sau đó dữ liệu này được tiến hành tiền xử lí.

```

RangeIndex: 16896 entries, 0 to 16895
Data columns (total 3 columns):
 #   Column   Non-Null Count   Dtype  
--- 
 0   field1    16896 non-null   float64
 1   field2    16894 non-null   float64
 2   field3    16894 non-null   float64
dtypes: float64(3)
memory usage: 396.1 KB
None
      field1      field2      field3
count  16896.000000  16894.000000  16894.000000
mean    28.194081    78.369471    0.055672
std     1.955180    7.882437    0.026399
min     0.000000    51.510000    0.000000
25%    26.780000    73.792500    0.040000
50%    27.930000    79.750000    0.060000
75%    29.500000    84.047500    0.070000
max    33.510000    94.290000    0.800000

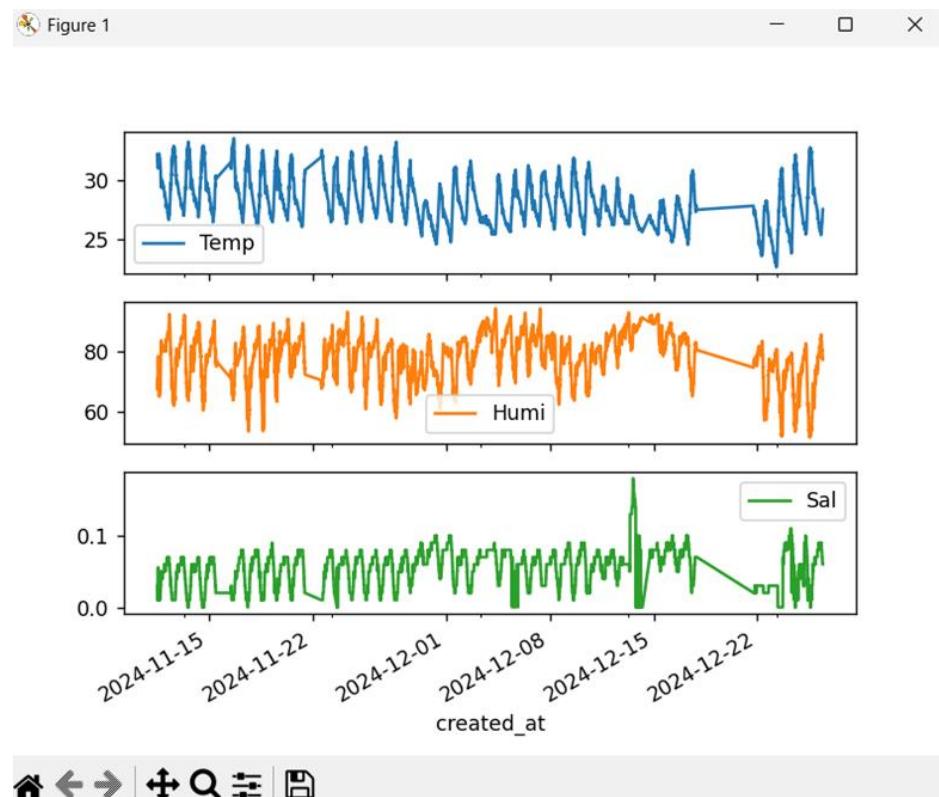
```

Hình 3.31 Tiền xử lí dữ liệu



Hình 3.32 Trực quan hóa bằng biểu đồ

Trực quan hóa thông qua biểu đồ để có cái nhìn tổng quát về dữ liệu. Dựa vào biểu đồ sẽ thấy dữ liệu sẽ có các điểm dữ liệu bị lỗi như nhiệt độ bằng 0 và độ mặn bằng 1, các điểm dữ liệu này sẽ được xử lý bằng cách gán cho chúng giá trị của điểm dữ liệu liền trước (do đây là dữ liệu liên tục theo thời gian, ta có thể xem như điểm dữ liệu bị lỗi là giá trị của điểm dữ liệu trước đó không thay đổi).



Hình 3.33 Sau khi xử lí nhiễu

Sau khi đã xử lý các điểm dữ liệu bị nhiễu thì sẽ tiến hành phân chia và chuẩn hóa. Tỉ lệ phân chia dữ liệu như sau: 60% huấn luyện, 30% đánh giá và 10% dùng để kiểm tra.

```

49      # chia bộ dữ liệu để huấn luyện
50      n = len(df)
51      train_df = df[0:int(n*0.6)]
52      val_df = df[int(n*0.6):int(n*0.9)]
53      test_df = df[int(n*0.9):]
54
55      # chuẩn hóa dữ liệu
56      train_mean = train_df.mean()
57      train_std = train_df.std()
58
59      train_df = (train_df - train_mean) / train_std
60      val_df = (val_df - train_mean) / train_std
61      test_df = (test_df - train_mean) / train_std
62
63      # Lưu train_mean và train_std dưới dạng NumPy arrays
64      train_mean.to_numpy().dump(os.path.join(DATA_DIR, 'train_mean.npy'))
65      train_std.to_numpy().dump(os.path.join(DATA_DIR, 'train_std.npy'))

```

Hình 3.34 Phân tách và chuẩn hóa dữ liệu

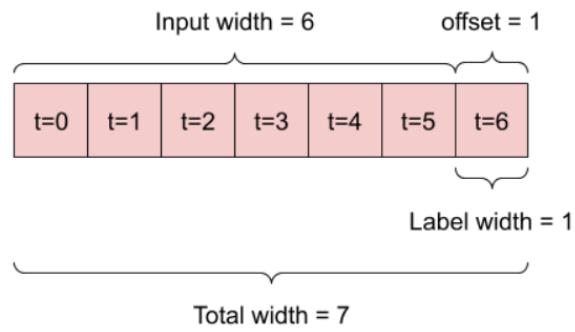
### 3.2.3.3 Phương pháp cửa sổ dữ liệu

Là kỹ thuật chia dữ liệu chuỗi thời gian thành các đoạn nhỏ liên tiếp, Mỗi cửa sổ chứa một số điểm dữ liệu liên tiếp theo thời gian, Cửa sổ "trượt" qua dữ liệu để tạo ra các mẫu huấn luyện

Các thành phần chính:

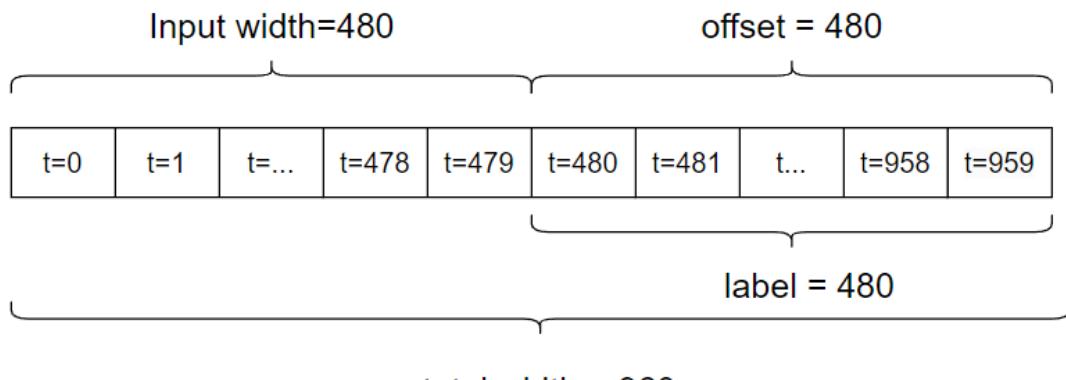
- Total width (kích thước cửa sổ): Số lượng điểm dữ liệu trong một cửa sổ
- Off set (bước trượt): Khoảng cách giữa các cửa sổ liên tiếp
- Label width: Số lượng điểm dữ liệu cần dự báo

Tùy thuộc vào bài toán mà có thể tạo ra nhiều loại cửa sổ khác nhau như mô hình chỉ dự đoán 1 điểm dữ liệu tiếp theo hoặc dự đoán nhiều điểm tiếp theo.



Hình 3.35 Cửa sổ với 1 đầu ra dựa trên 6 giá trị lịch sử

Đối với mô hình của chúng ta sẽ tiến hành dữ đoán nhiều điểm dữ liệu tiếp theo. Dữ liệu được thu thập cách nhau mỗi 3 phút nên 1 ngày sẽ có 480 mẫu dữ liệu. Dùng dữ liệu 1 ngày dự đoán 1 ngày tiếp theo, vậy cửa sổ sẽ như sau:



Hình 3.36 Cửa sổ sử dụng cho mô hình

Sau đó các dữ liệu này sẽ được đưa qua một class WindowGenerator để tạo thành các cửa sổ dùng cho việc huấn luyện mô hình.

```

7  class WindowGenerator:
8      def __init__(self, input_width, label_width, shift,
9                   train_df, val_df, test_df,
10                  label_columns=None):
11          # Store the raw data.
12          self.train_df = train_df
13          self.val_df = val_df
14          self.test_df = test_df
15
16          # Work out the label column indices.
17          self.label_columns = label_columns
18          if label_columns is not None:
19              self.label_columns_indices = {name: i for i, name in enumerate(label_columns)}
20              self.column_indices = {name: i for i, name in enumerate(train_df.columns)}
21
22          # Window parameters
23          self.input_width = input_width
24          self.label_width = label_width
25          self.shift = shift
26
27          self.total_window_size = input_width + shift
28
29          self.input_slice = slice(0, input_width)
30          self.input_indices = np.arange(self.total_window_size)[self.input_slice]
31
32          self.label_start = self.total_window_size - self.label_width
33          self.labels_slice = slice(self.label_start, None)
34          self.label_indices = np.arange(self.total_window_size)[self.labels_slice]
35
36          self.num_features = train_df.shape[1]
37
38      def split_window(self, features):
39          inputs = features[:, self.input_slice, :]
40          labels = features[:, self.labels_slice, :]
41          if self.label_columns is not None:
42              labels = tf.stack(

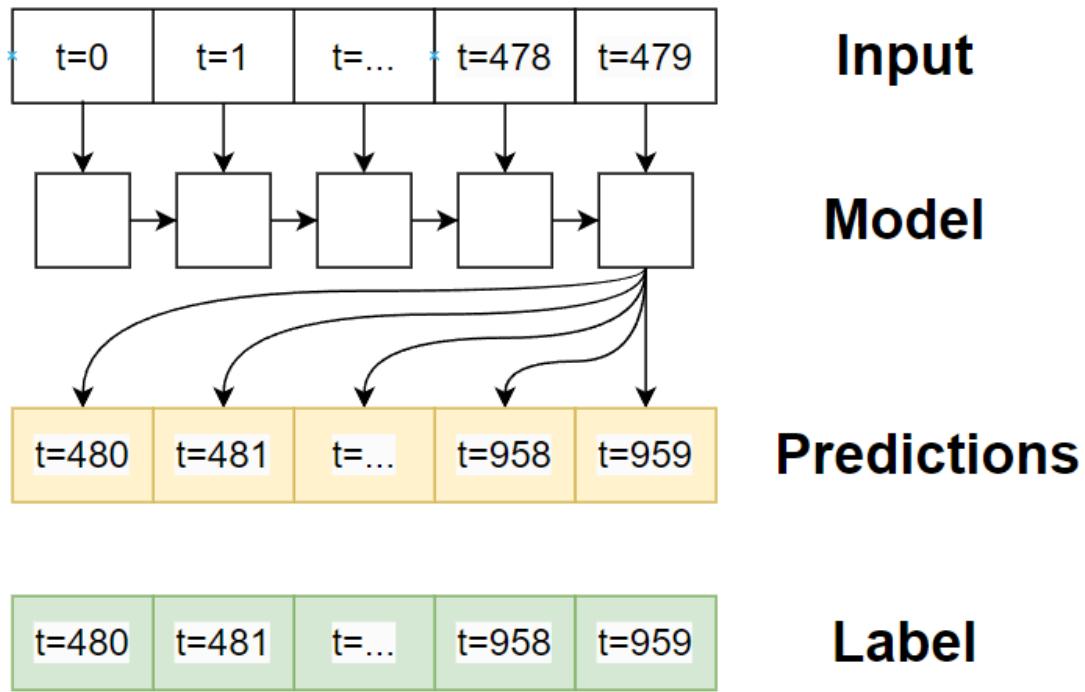
```

Hình 3.37 Lớp WindowGenerator

#### 3.2.3.4 Xây dựng và Huấn luyện mô hình

Nhiều mô hình phổ biến sử dụng trong dự đoán dữ liệu chuỗi thời gian như: LSTM, GRU, CNN, Transformer, Prophet...

Dựa trên các đặc điểm của bộ dữ liệu và yêu cầu bài toán như dữ liệu có phụ thuộc thời gian dài hạn, chuỗi thời gian có độ dài trung bình, có đủ dữ liệu huấn luyện, yêu cầu độ chính xác cao... nhóm quyết định lựa chọn mô hình LSTM.



Hình 3.38 Cấu trúc mô hình

### 3.2.3.5 Khởi tạo mô hình và Huấn luyện

```

12 # Khởi tạo model
13 def create_lstm_model(input_width, num_features, out_steps):
14     return tf.keras.Sequential([
15         tf.keras.layers.LSTM(32, return_sequences=False, input_shape=(input_width, num_features)),
16         tf.keras.layers.Dense(out_steps*num_features,
17             kernel_initializer=tf.initializers.zeros()),
18         tf.keras.layers.Reshape([out_steps, num_features])
19     ])
20
21 # Huấn Luyện model
22 def train_lstm_model(window, epochs=20):
23     model = create_lstm_model(input_width=window.input_width, num_features=window.num_features, out_steps=window.label_width)
24     model.summary()
25     model.compile(loss=tf.keras.losses.MeanSquaredError(),
26                   optimizer=tf.keras.optimizers.Adam(),
27                   metrics=[tf.keras.metrics.MeanAbsoluteError()])
28
29     history = model.fit(window.train, epochs=epochs,
30                         validation_data=window.val,
31                         callbacks=[tf.keras.callbacks.EarlyStopping(patience=3, mode='min')])
32 # Sau khi huấn luyện thì lưu lại kết quả dưới dạng file keras
33 model.save(os.path.join(MODEL_DIR, 'lstm_model.keras'))
34 return model

```

Hình 3.39 Khởi tạo mô hình

Mô hình xây dựng có kích thước cửa sổ đầu vào là  $480 \times 3$  tương ứng với dữ liệu 1 ngày là 480 điểm và 3 đặc trưng là Nhiệt độ, Độ ẩm và Độ mặn, lớp đầu vào của mô hình có 32 nơ ron. Tiếp theo là lớp ẩn với số nơ ron là kích thước đầu ra nhân với số

đặc trưng dự đoán là  $480 \times 3 = 1440$ . Cuối cùng là lớp đầu ra với kích thước là  $480 \times 3$  tương ứng với giá trị dự đoán mỗi ngày cho từng đặc trưng dữ liệu.

Model: "sequential"		
Layer (type)	Output Shape	Param #
lstm (LSTM)	(None, 32)	4,608
dense (Dense)	(None, 1440)	47,520
reshape (Reshape)	(None, 480, 3)	0

Total params: 52,128 (203.62 KB)  
Trainable params: 52,128 (203.62 KB)  
Non-trainable params: 0 (0.00 B)

Hình 3.40 Kiến trúc Mô hình LSTM

Tổng tham số của mô hình là 52 128

Mô hình sử dụng hàm mất mát MSE (Mean Squared Error) để đo lường sai số bình phương trung bình giữa giá trị thực tế và giá trị dự đoán. Quá trình cập nhật trọng số trong khi huấn luyện được thực hiện thông qua bộ tối ưu hóa Adam. Ngoài ra, Mean Absolute Error (MAE) được áp dụng để đánh giá hiệu quả của mô hình, nhờ khả năng đo độ chênh lệch trung bình theo giá trị tuyệt đối giữa dự đoán và thực tế, mà không bị ảnh hưởng bởi các lỗi bất thường lớn.

Mô hình được huấn luyện trên tập dữ liệu huấn luyện (train) và đánh giá trên tập dữ liệu kiểm tra (val) trong khoảng 20 epochs. Cơ chế dừng sớm (early stopping) được áp dụng để ngăn chặn hiện tượng overfitting.

## Huấn luyện lần 1

```
Epoch 1/20  
287/287 22s 74ms/step - loss: 0.7713 - mean_absolute_error: 0.6980 - val_loss: 1.0564 - val_mean_absolute_error: 0.7654  
Epoch 2/20  
287/287 20s 71ms/step - loss: 0.4519 - mean_absolute_error: 0.5025 - val_loss: 0.9476 - val_mean_absolute_error: 0.7003  
Epoch 3/20  
287/287 20s 70ms/step - loss: 0.3794 - mean_absolute_error: 0.4488 - val_loss: 0.9182 - val_mean_absolute_error: 0.6725  
Epoch 4/20  
287/287 20s 69ms/step - loss: 0.3252 - mean_absolute_error: 0.4106 - val_loss: 0.9054 - val_mean_absolute_error: 0.6519  
Epoch 5/20  
287/287 21s 74ms/step - loss: 0.3080 - mean_absolute_error: 0.3959 - val_loss: 0.8612 - val_mean_absolute_error: 0.6401  
Epoch 6/20  
287/287 20s 71ms/step - loss: 0.3272 - mean_absolute_error: 0.4073 - val_loss: 0.8713 - val_mean_absolute_error: 0.6415  
Epoch 7/20  
287/287 20s 70ms/step - loss: 0.2791 - mean_absolute_error: 0.3774 - val_loss: 0.7776 - val_mean_absolute_error: 0.6027  
Epoch 8/20  
287/287 20s 69ms/step - loss: 0.2664 - mean_absolute_error: 0.3651 - val_loss: 0.8582 - val_mean_absolute_error: 0.6310  
Epoch 9/20  
287/287 20s 69ms/step - loss: 0.2872 - mean_absolute_error: 0.3766 - val_loss: 0.9733 - val_mean_absolute_error: 0.6758  
Epoch 10/20  
287/287 20s 69ms/step - loss: 0.3071 - mean_absolute_error: 0.3918 - val_loss: 0.9328 - val_mean_absolute_error: 0.6571  
--- 204.5916087627411 seconds ---  
129/129 3s 19ms/step - loss: 0.8933 - mean_absolute_error: 0.6443
```

Hình 3.41 Huấn luyện lần 1

## Huấn luyện lần 2

```
Epoch 1/20  
287/287 20s 63ms/step - loss: 0.8025 - mean_absolute_error: 0.7132 - val_loss: 1.0935 - val_mean_absolute_error: 0.7825  
Epoch 2/20  
287/287 18s 63ms/step - loss: 0.4860 - mean_absolute_error: 0.5222 - val_loss: 1.0126 - val_mean_absolute_error: 0.7289  
Epoch 3/20  
287/287 18s 63ms/step - loss: 0.4000 - mean_absolute_error: 0.4632 - val_loss: 1.0092 - val_mean_absolute_error: 0.7204  
Epoch 4/20  
287/287 18s 63ms/step - loss: 0.3346 - mean_absolute_error: 0.4211 - val_loss: 0.9426 - val_mean_absolute_error: 0.6805  
Epoch 5/20  
287/287 18s 63ms/step - loss: 0.3432 - mean_absolute_error: 0.4215 - val_loss: 1.0033 - val_mean_absolute_error: 0.7083  
Epoch 6/20  
287/287 18s 63ms/step - loss: 0.3801 - mean_absolute_error: 0.4472 - val_loss: 0.9902 - val_mean_absolute_error: 0.7010  
Epoch 7/20  
287/287 19s 65ms/step - loss: 0.4078 - mean_absolute_error: 0.4655 - val_loss: 0.9985 - val_mean_absolute_error: 0.6943  
--- 129.72345566749573 seconds ---  
129/129 2s 19ms/step - loss: 1.0244 - mean_absolute_error: 0.7025
```

Hình 3.42 Huấn luyện lần 2

## Huấn luyện lần 3

```
Epoch 1/20  
287/287 24s 79ms/step - loss: 0.7865 - mean_absolute_error: 0.7050 - val_loss: 1.1333 - val_mean_absolute_error: 0.8034  
Epoch 2/20  
287/287 24s 84ms/step - loss: 0.4959 - mean_absolute_error: 0.5262 - val_loss: 1.1008 - val_mean_absolute_error: 0.7874  
Epoch 3/20  
287/287 20s 71ms/step - loss: 0.4197 - mean_absolute_error: 0.4755 - val_loss: 1.0693 - val_mean_absolute_error: 0.7618  
Epoch 4/20  
287/287 21s 72ms/step - loss: 0.4318 - mean_absolute_error: 0.4820 - val_loss: 1.1405 - val_mean_absolute_error: 0.8026  
Epoch 5/20  
287/287 20s 70ms/step - loss: 0.4970 - mean_absolute_error: 0.5219 - val_loss: 1.0347 - val_mean_absolute_error: 0.7388  
Epoch 6/20  
287/287 20s 71ms/step - loss: 0.3926 - mean_absolute_error: 0.4542 - val_loss: 1.1130 - val_mean_absolute_error: 0.7616  
Epoch 7/20  
287/287 21s 71ms/step - loss: 0.3932 - mean_absolute_error: 0.4561 - val_loss: 1.0555 - val_mean_absolute_error: 0.7461  
Epoch 8/20  
287/287 20s 70ms/step - loss: 0.3294 - mean_absolute_error: 0.4113 - val_loss: 1.0387 - val_mean_absolute_error: 0.7371  
--- 171.0495159626007 seconds ---  
129/129 3s 21ms/step - loss: 1.0235 - mean_absolute_error: 0.7322
```

Hình 3.43 Huấn luyện lần 3

### 3.2.3.6 Kiểm tra và Đánh giá.

Về thời gian huấn luyện, trung bình cho 3 lần huấn luyện mỗi lần mất khoảng 3 phút. Tiếp theo dùng MAE để đánh giá hiệu suất của mô hình do MAE đo độ lớn trung bình sai số tuyệt đối. Kết quả MAE chạy trên tập dữ liệu kiểm tra ở cả 3 lần.

Lần 1:

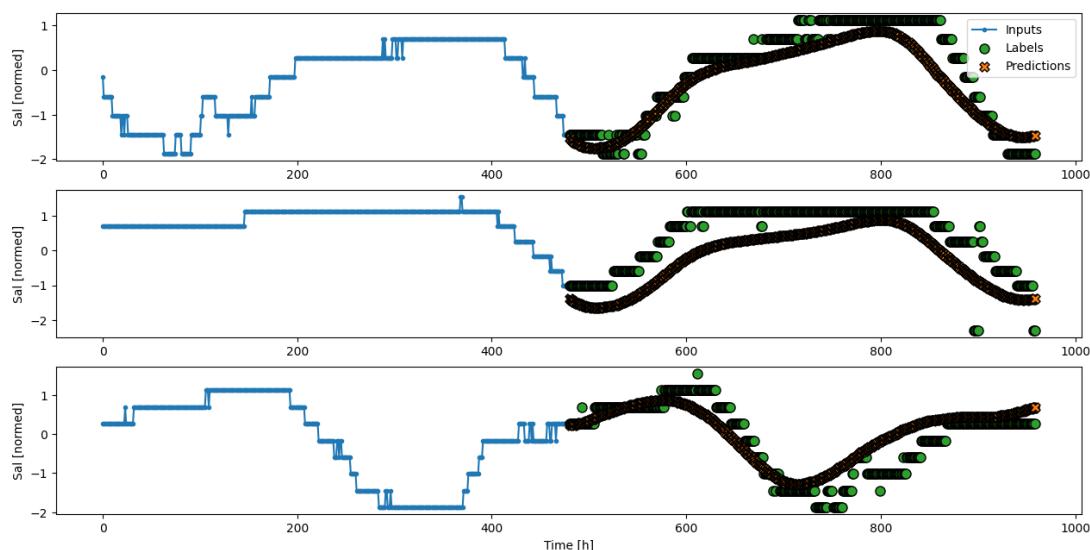
```
MAE Val: [0.9328470230102539, 0.6570720672607422]
MAE Test: [0.2811008095741272, 0.37276917695999146]
```

Lần 2:

```
MAE Val: [0.9984517693519592, 0.6943078637123108]
MAE Test: [0.327779084444046, 0.407668799161911]
```

Lần 3:

```
MAE Val: [1.038682460784912, 0.7371185421943665]
MAE Test: [0.3080398142337799, 0.3930746912956238]
```



Hình 3. 3.444 Kết quả kiểm tra với đặc trưng độ mặn

Dựa vào kết quả MAE trên tập dữ liệu kiểm tra ở cả 3 lần thì MAE kiểm tra đều xấp xỉ với MAE huấn luyện cho thấy mô hình đã được tối ưu tốt và có khả năng tổng quát hóa cao. Nhưng tỉ lệ vẫn còn cao vào khoảng 0.4. Vì thế cần cải tiến mô hình bằng cách chia tăng số nơ ron đầu vào lên 128.

Sau khi tăng số đầu vào lên 128 kiến trúc mô hình sẽ như sau:

Model: "sequential"		
Layer (type)	Output Shape	Param #
lstm (LSTM)	(None, 128)	67,584
dense (Dense)	(None, 1440)	185,760
reshape (Reshape)	(None, 480, 3)	0

Total params: 253,344 (989.62 KB)  
Trainable params: 253,344 (989.62 KB)  
Non-trainable params: 0 (0.00 B)

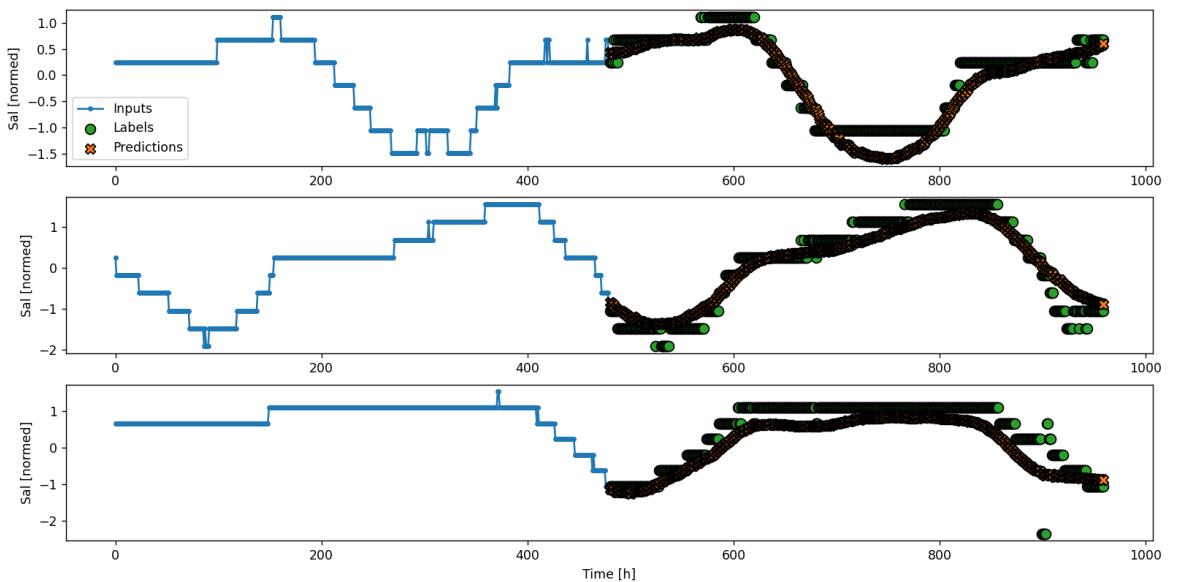
Hình 3.45 Kiến trúc mô hình LSTM sau khi cải tiến

Mô hình sẽ có tổng cộng 253 344 tham số.

```
Epoch 1/20
340/340 49s 141ms/step - loss: 0.6638 - mean_absolute_error: 0.6299 - val_loss: 1.6731 - val_mean_absolute_error: 0.9733
Epoch 2/20
340/340 50s 148ms/step - loss: 0.3299 - mean_absolute_error: 0.4147 - val_loss: 1.2583 - val_mean_absolute_error: 0.8028
Epoch 3/20
340/340 49s 143ms/step - loss: 0.2936 - mean_absolute_error: 0.3836 - val_loss: 1.2117 - val_mean_absolute_error: 0.7743
Epoch 4/20
340/340 53s 156ms/step - loss: 0.2996 - mean_absolute_error: 0.3861 - val_loss: 1.2119 - val_mean_absolute_error: 0.7651
Epoch 5/20
340/340 54s 160ms/step - loss: 0.2738 - mean_absolute_error: 0.3678 - val_loss: 1.4596 - val_mean_absolute_error: 0.8558
Epoch 6/20
340/340 54s 160ms/step - loss: 0.2699 - mean_absolute_error: 0.3648 - val_loss: 1.2901 - val_mean_absolute_error: 0.7897
--- 311.24129366874695 seconds ---
76/76 5s 58ms/step - loss: 1.2854 - mean_absolute_error: 0.7862
MAE Val: [1.2900760173797607, 0.7896864414215088]
MAE Test: [0.2246905118227005, 0.33348768949508667]
```

Hình 3.46 Huấn luyện sau khi cải tiến

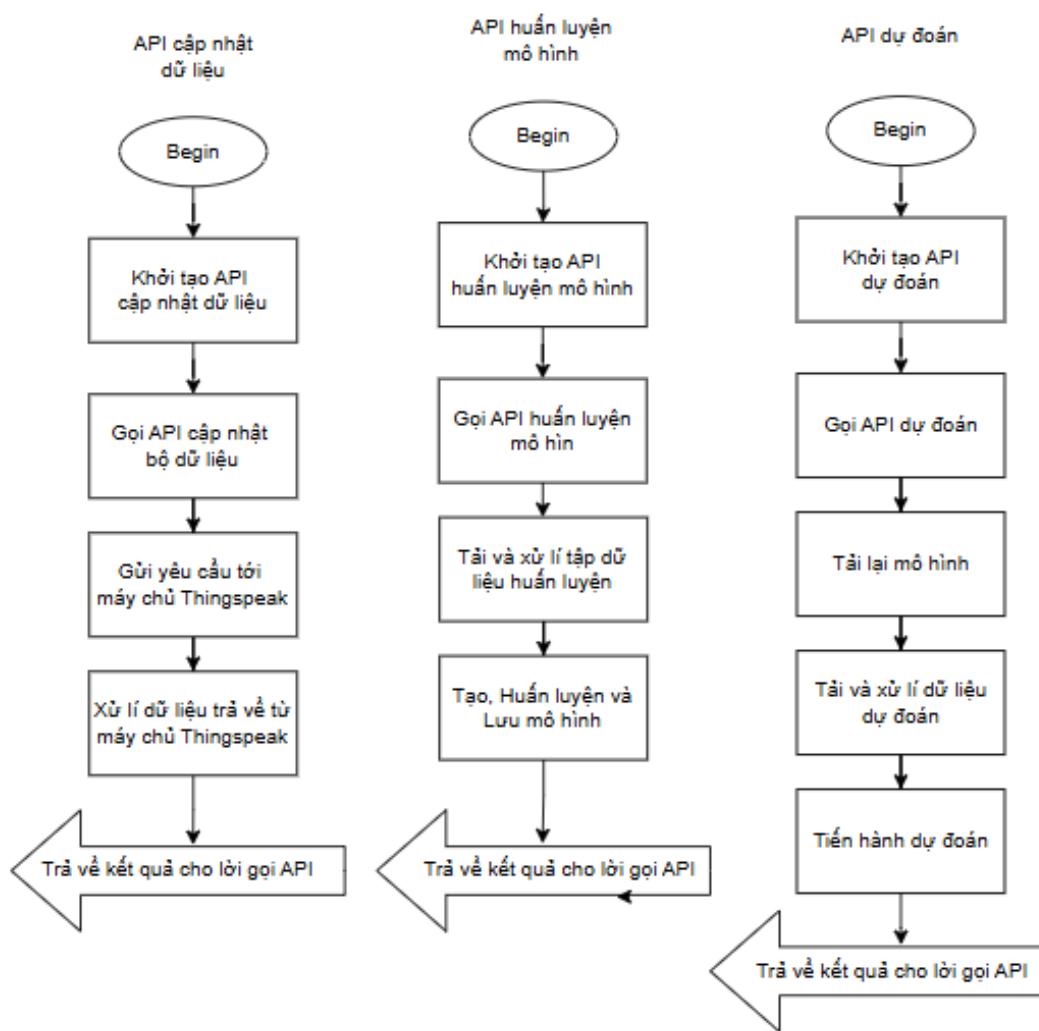
Sau khi tăng thêm nơ ron thì mô hình tăng thêm gian huấn luyện lên hơn 5 phút nhưng MAE trên dữ liệu kiểm tra sẽ được cải thiện giảm xuống thấp hơn cụ thể là 0.33 so với trước đó thấp nhất là 0.37.



Hình 3.47 Kết quả kiểm tra với đặc trưng độ mặn sau khi cải tiến

Kết quả trực quan bằng biểu đồ cũng cho thấy kết quả dự đoán tốt hơn so với trước đó. Sau khi kết quả đã được cải thiện, các bước trên sẽ được chuyển đổi thành các API và chạy trên server để có thể tích hợp với ứng dụng mobile.

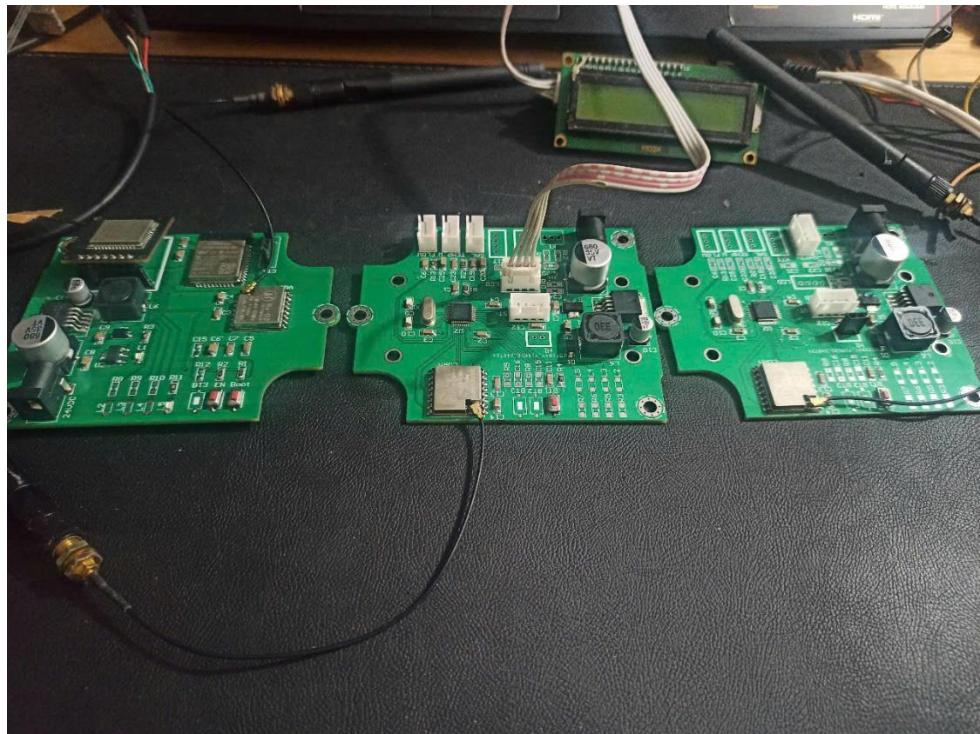
Khi bắt đầu, máy chủ FastAPI sẽ khởi tạo các API sau đó lắng nghe các yêu cầu từ các máy khách, nếu nhận được yêu cầu cập nhật bộ dữ liệu huấn luyện thành các dữ liệu mới nhất thì máy chủ FastAPI sẽ tiến hành gửi yêu cầu tới máy chủ Thingspeak để lấy dữ liệu, sau đó máy chủ FastAPI sẽ tiến hành xử lí, cập nhật các dữ liệu này và trả về thông báo cho máy khách biết. Nếu yêu cầu nhận được là huấn luyện mô hình thì máy chủ FastAPI sẽ tải bộ dữ liệu huấn luyện đang lưu trữ và tiến hành xử lí để chuẩn bị cho việc huấn luyện, tiếp theo sẽ tạo mô hình và tiến hành huấn luyện, sau khi huấn luyện xong thì sẽ tiến hành lưu lại tham số cho mô hình và trả về thông báo cho máy khách. Nếu nhận được yêu cầu dự đoán, máy chủ sẽ tải lại mô hình đã huấn luyện và lưu trước đó, tiếp theo tải lên những mẫu dữ liệu mới nhất và xử lí chúng để đưa vào cho mô hình dự đoán, sau đó tiến hành dự đoán và trả về kết quả cho máy khách.



Hình 3. 3.48 Lưu đồ chương trình AI dự đoán tương lai

## CHƯƠNG 4 THỰC NGHIỆM VÀ ĐÁNH GIÁ KẾT QUẢ

### 4.1 Kết quả phần cứng



Hình 4.1 Hàn linh kiện lên mạch in



Hình 4.2 Test thiết bị sau lúc đóng hộp



Hình 4.3 Đóng gói các mạch vào hộp chống nước



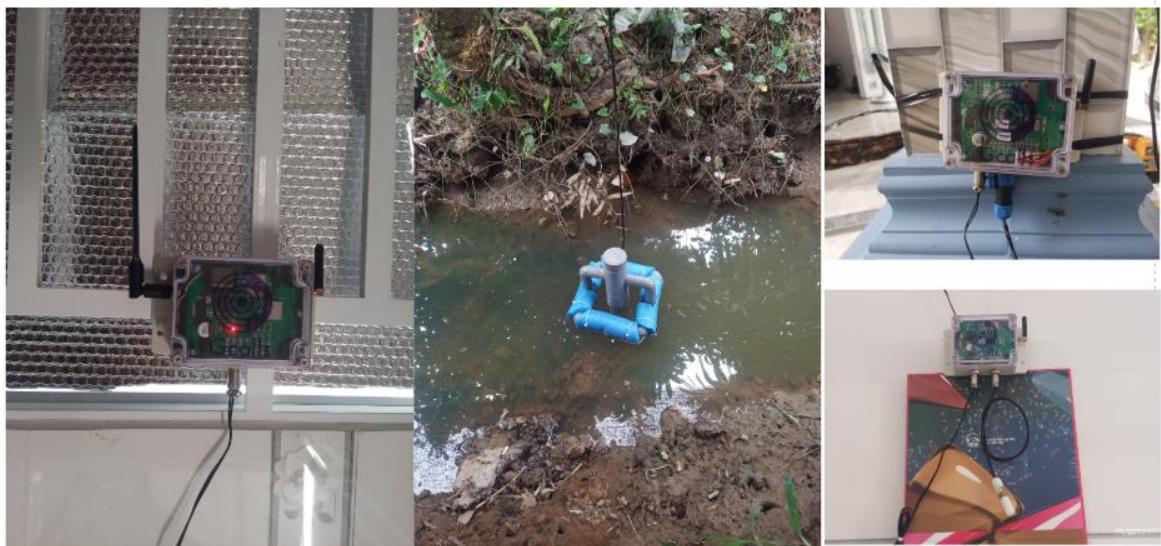
Hình 4.4 Gateway sau khi đóng gói



Hình 4.5 Node sau khi đóng gói

## 4.2 Lắp đặt thực tế

Sau khi thiết kế và đóng gói sản phẩm, thì nhóm được đi đến vườn trái cây ở Tiền Giang để lắp đặt thu thập những dữ liệu từ thực tế.



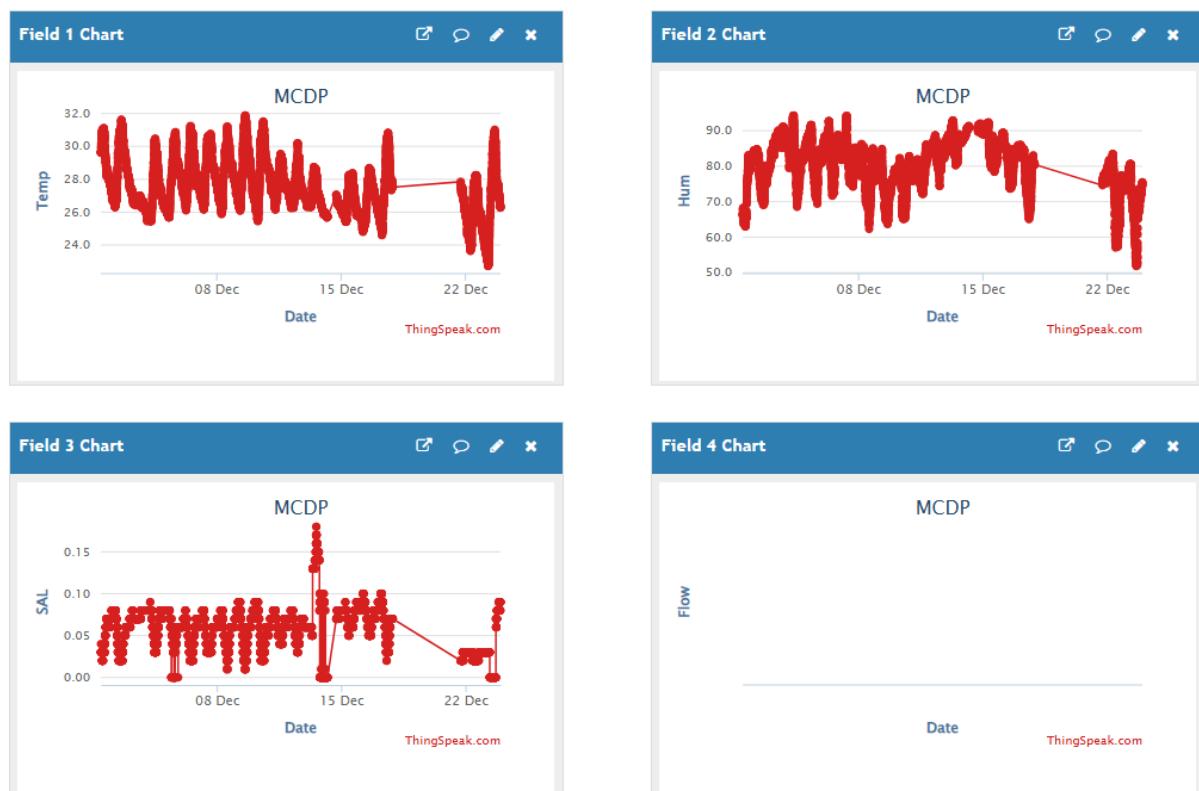
Hình 4.6 Hình ảnh lắp đặt thực tế

Hình ảnh này, được nhóm đi về tỉnh miền tây để thực nghiệm. Cảm biến đo độ mặn được bơ vào trong phao, phao này do nhóm tự làm thủ công từ ống nhựa và phao bơi dạng ống. Các node được chủ vườn yêu cầu, lắp ở các vị trí do chủ nhà yêu cầu, anten

dùng thực nghiệm là 3db cho LoRa và anten cho 4G. Hệ thống được cấp điện từ adapter 24V.

### 4.3 Kết quả phần mềm

#### 4.3.1 Server



Hình 4.7 Hình ảnh dữ liệu được lấy được từ thiết bị IOT

Hệ thống được lắp đặt thu thập dữ liệu từ ngày 11/11/2024 cho đến nay. Dữ liệu đã có tính chu kỳ, có 1 số sự cố gãy phải như gãy phai mất điện, mất wifi nên có 1 vài đoạn dữ liệu không được liên tục.

#### 4.3.2 App

Sau khi có dữ liệu trên server thực hiện cài đặt app để theo dõi thông số.



Hình 4.8 Giao diện login của app

Nếu nhập tài khoản, mật khẩu không đúng thì sẽ trả về kết quả “login fail”.

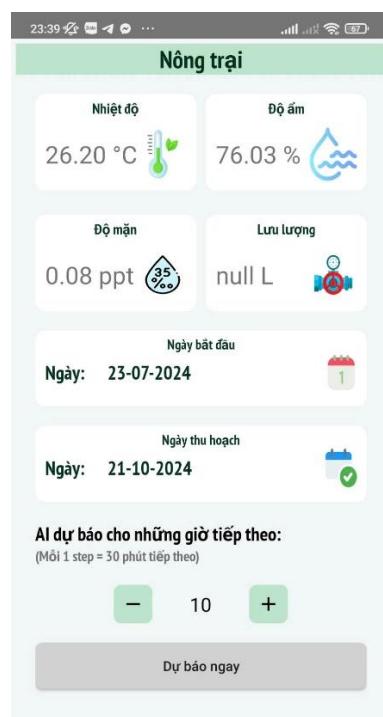


Hình 4.9 Giao diện khi thông tin đăng nhập sai

Khi đăng nhập thành công vào app, thì giao diện chuyển sang trang chủ sẽ có các trang trại của tài khoản người dùng, ảnh đại diện người dùng, thông báo, ngày hiện tại.



Hình 4.10 Quản lý các nông trại



Hình 4.11 Các giá trị hiện tại của cảm biến

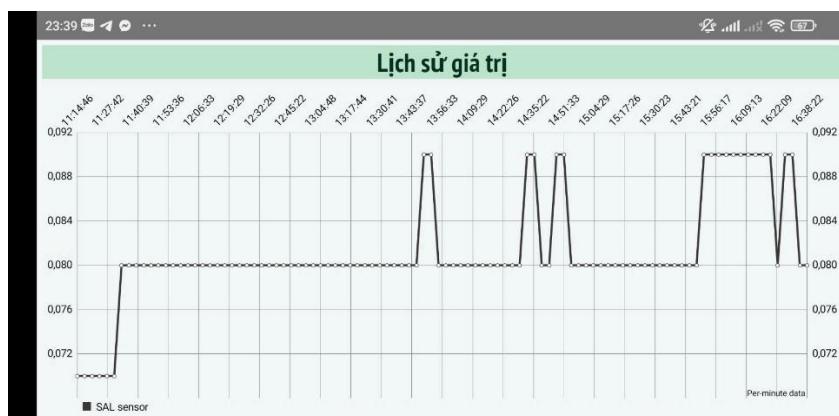
Sau khi nhấn vào mục trang trại “MCDP farm” thì giao diện chuyển vào giao diện quản lý giá trị cảm biến thu thập được. Khi nhấn vào từng module cảm biến thì sẽ ra lịch sử giá trị của từng cảm biến.



Hình 4.12 Biểu đồ giá trị của nhiệt độ



Hình 4.13 Biểu đồ giá trị của độ ẩm



Hình 4.14 Biểu đồ giá trị của độ mặn

Phần dự đoán có thể điều chỉnh số giá trị muốn xem trong tương lai. Nhấn vào dấu “+” để tăng đơn vị mỗi lần tăng là 10 giá trị, nhấn “-” để giảm giá trị số điểm dự đoán trong tương lai đơn vị mỗi lần giảm là 10. Nhấn “Dự đoán ngay” để xem dữ liệu dự đoán trong tương lai.



Hình 4.15 Biểu đồ giá trị sau khi dự đoán

#### 4.4 Đánh giá kết quả

Hệ thống đã hoàn thành hầu hết hoàn thành yêu cầu cho hệ thống. Độ chính xác của cảm biến chưa được so sánh với máy đo chuyên dụng. Dữ liệu tần suất 3 phút 1 gói tin lên server. Khả năng chống nước, chống ẩm của sản phẩm minh chứng trong thời gian lắp đặt ở tại vườn. Phần mềm app người dùng cũng hoạt động giám sát tốt trong thời gian lắp đặt thử nghiệm.

Mức tiêu thụ của hệ thống khi dùng mức điện áp 12V, node có gắn cảm biến SHT30 lúc chưa hoạt động thì dòng điện là 31mA, lúc hoạt động dòng điện là 32mA. Đối với node có gắn cảm biến đo độ mặn và cảm biến lưu lượng, chưa hoạt động đang ở chế độ sleep dòng điện là 47mA và khi hoạt động dòng là 68mA. Đối với gateway, dùng điện áp 12V thì dòng điện khi ở chế độ sleep là 9mA và khi hoạt động thì dòng điện tiêu thụ là 68mA.

## KẾT LUẬN VÀ KIẾN NGHỊ

### 4.1 Kết luận

Hệ thống chạy thực tế trong gần 2 tháng và đạt được các mục tiêu đã đề ra trước đó.

#### Về các node và gateway:

- Đã xây dựng được các node để lấy các giá trị độ mặn, độ ẩm, nhiệt độ và lượng nước.
- Đã có các module LoRa được trang bị trên các board từ node đến gateway.
- Có 2 giải pháp để kết nối mạng gửi dữ liệu về server là bằng 4G và Wifi.
- Truyền nhận dữ liệu từ gateway với node thì trong thời gian qua đang rất ổn định.

#### Về app cho sử dụng:

- Đưa ra được gateway lấy được dữ liệu và gửi đến server ThingSpeak.
- Các node và gateway tự liên kết với nhau tạo thành mạng lưới.
- Có app để theo dõi giá trị đó được và đồng thời app này có thể xem được kết quả dự đoán giá trị độ mặn cho tương lai.
- Phần mềm có tiêu chuẩn bảo mật dữ liệu bằng hình thức đăng nhập thông dụng dùng id người dùng và mật mã được cấp.

Xây dựng thành công mô hình AI có khả năng dự đoán các giá trị tương lai, cải tiến được mô hình để giúp tăng độ chính xác của kết quả dự đoán, triển khai thành công mô hình thành các API chạy trên server để có thể luôn cập nhật mô hình với dữ liệu mới nhất và giao tiếp với các ứng dụng trong hệ thống.

Ngoài những mặt thuận lợi, thì còn những mặt hạn chế sau:

- Chưa đủ chi phí để mua những máy đo trên thị trường để so sánh độ chính xác của giá trị cảm biến.
- Server đang dùng của thingspeak cung cấp miễn phí nên còn 1 số giới hạn trong việc cài đặt.
- Không có server public để setup chi AI nên còn 1 số hạn chế về phần dự đoán các giá trị của cảm biến trong tương lai.
- Công suất tiêu thụ của thiết bị quá lớn, nếu dùng pin trong điều kiện không lý tưởng chỉ được vài ngày.

## 4.2 Kiến nghị

Các phần trong hệ thống cần cải thiện và phát triển:

### Về phần cứng:

- Phải kiểm tra so với thiết bị đo chuyên dụng xem sai số cảm biến.
- Phát triển thêm về phần xóa thiết bị ra khỏi hệ thống.
- Phát triển thêm GPS cho hệ thống để xác định được vị trí.
- Về phần năng lượng hiện tại công suất tiêu thụ khá lớn, muốn duy trì được vài tháng phải thay thế các IC có dòng điện tĩnh là vài  $\mu$ A.

### Về phần mềm:

- Cần phát triển 1 server public riêng để cài đặt các thông số dễ dàng hơn.
- Về phần AI nên được đưa lên server public để có thể gọi cho dễ dàng.
- Phát triển thêm phần cài 1 thiết bị mới vào mạng lưới bằng app.

## TÀI LIỆU THAM KHẢO

- [1] T. C. T. KÊ, "TỔNG CỤC THỐNG KÊ," BÁO CÁO TÌNH HÌNH KINH TẾ – XÃ HỘI QUÝ IV VÀ NĂM 2023, 2023. [Trực tuyến]. Available: <https://www.gso.gov.vn/bai-top/2023/12/bao-cao-tinh-hinh-kinh-te-xa-hoi-quy-iv-va-nam-2023/>.
- [2] T. TRÀ, "GDP ngành nông nghiệp tăng trưởng cao nhất trong 10 năm gần đây," Báo Nhân Dân điện tử, 2023.
- [3] B. T. n. v. M. Trường, "Báo cáo hiện trạng môi trường quốc gia năm 2018 Chuyên đề: Môi trường nước lưu vực sông," Bộ Tài Nguyên và Môi Trường, 2019.
- [4] H. Đạt, "Dự báo xâm nhập mặn đến sớm ở Đồng bằng sông Cửu Long," baotintuc.vn, 2023.
- [5] T. Chung, "45% diện tích ĐBSCL có thể nhiễm mặn vào năm 2030," Báo Điện tử Chính phủ, 2016.
- [6] "ORACLE," What Is the Internet of Things?, [Trực tuyến]. Available: <https://www.oracle.com/vn/internet-of-things/>. [Đã truy cập 29 12 2024].
- [7] "Amazon Web Services," What Is Artificial Intelligence (AI)?, [Trực tuyến]. Available: <https://s.net.vn/EMWA>. [Đã truy cập 29 12 2024].
- [8] A. I. Al-Alawi, "WiFi Technology: Future Market Challenges and Opportunities," *Journal of Computer Science* 2 (1): 13-18, 2006.
- [9] T. Weber, "Emnify," What Is 4G? Fourth Generation Cellular Networks Explained, 22 04 2021. [Trực tuyến]. Available: <https://www.emnify.com/iot-glossary/4g#authorbio>.
- [10] Semtech, "Semtech," LoRa Connect™ 137MHz to 1020MHz Long Range Low Power Transceiver, [Trực tuyến]. Available: <https://www.semtech.com/products/wireless-rf/lora-connect/sx1276>.
- [11] K. LE, "TAPIT," Tiếp cận với công nghệ truyền thông không dây LoRa, 2019. [Online]. Available: <https://tapit.vn/tiep-can-voi-cong-nghe-truyen-thong-khong-day-lora/>.
- [12] B. T. T. V. T. THÔNG, "Thông tư số 46/2016/TT-BTTTT," Bộ trưởng Bộ Thông tin và Truyền thông, 2019.
- [13] N. M. Ngọc, "Tài liệu Bài giảng hệ thống nhúng," Khoa CNDT, 2024.
- [14] T. T. Tính, Giáo Trình thổ nhưỡng, Khoa Nông Nghiệp và Sinh học Úng dụng, Trường Đại học Cần Thơ.