

**ĐẠI HỌC QUỐC GIA HÀ NỘI  
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ**



**Trần Hữu Nam**

**NGHIÊN CỨU, PHÁT TRIỂN THIẾT BỊ XÁC  
ĐỊNH TỰ THẾ NGỦ SỬ DỤNG CẢM BIẾN GIA  
TỐC**

**LUẬN VĂN THẠC SĨ**

**HÀ NỘI - 2025**

ĐẠI HỌC QUỐC GIA HÀ NỘI  
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

Trần Hữu Nam

NGHIÊN CỨU, PHÁT TRIỂN THIẾT BỊ XÁC  
ĐỊNH TƯ THẾ NGỦ SỬ DỤNG CẢM BIẾN GIA  
TỐC

Ngành: Điện tử viễn thông

Chuyên ngành: Kỹ thuật điện tử

Mã số: 23025029

LUẬN VĂN THẠC SĨ

NGƯỜI HƯỚNG DẪN KHOA HỌC: PGS.TS. Mai Anh Tuấn

HÀ NỘI - 2025

## LỜI CAM ĐOAN

Tôi xin cam đoan: đề tài luận văn thạc sĩ “Nghiên cứu, phát triển thiết bị xác định tư thế ngủ sử dụng cảm biến gia tốc” là công trình nghiên cứu của tôi dưới sự hướng dẫn của thầy **PGS. TS. Mai Anh Tuấn** và thầy **ThS. Trần Ngọc Thái** cùng với sự tham khảo từ những tài liệu đã liệt kê trong mục Tài liệu tham khảo. Tôi không sao chép công trình nghiên cứu của cá nhân khác dưới bất kỳ hình thức nào. Nếu có tôi xin hoàn toàn chịu trách nhiệm.

Hà Nội, ngày tháng năm 2025  
Học viên

Trần Hữu Nam

## **LỜI CẢM ƠN**

Lời đầu tiên, tôi xin gửi lời cảm ơn đến thầy PGS.TS. Mai Anh Tuấn và thầy ThS. Trần Ngọc Thái vì đã tận tình hướng dẫn, truyền đạt kiến thức cho tôi trong suốt quá trình học tập và thực hiện đề tài. Tôi xin cảm ơn tập thể thầy, cô khoa “Điện tử Viễn thông”, Trường Đại học Công nghệ - ĐHQGHN, đã giảng dạy tôi trong quá trình tôi học tập tại trường. Tôi cũng xin cảm ơn các anh chị ở Bộ môn Công nghệ Vi cơ Điện tử và Kỹ thuật Máy tính đã tạo điều kiện giúp đỡ, chỉ bảo tôi trong thời gian làm khóa luận. Cuối cùng, tôi xin cảm ơn bố mẹ, gia đình cũng như bạn bè, tập thể lớp K30 đã luôn đồng hành, chia sẻ và động viên tôi suốt thời gian qua.

## TÓM TẮT

**Tóm tắt:** Ngưng thở tắc nghẽn khi ngủ (Obstructive Sleep Apnea - OSA) là tình trạng rối loạn hô hấp khi ngủ thường gặp, được đặc trưng bởi những cơn ngưng/giảm thở ngắn, lặp lại trong khi ngủ, gây gián đoạn giấc ngủ bởi hành vi thức giấc thường xuyên và giảm oxy máu ngắt quãng[Epstein2009]. Nghiên cứu[Salari2025] đã phân tích dữ liệu từ 15 nghiên cứu với tổng cộng 42.924 người. Kết quả cho thấy tỷ lệ mắc chứng buồn ngủ ban ngày quá mức (EDS) ở bệnh nhân ngưng thở tắc nghẽn khi ngủ (OSA) trên toàn cầu là 39,9% (khoảng tin cậy 95%: 34,4%–45,7%).

Trong bối cảnh công nghệ chế tạo ngày càng phát triển mạnh mẽ, việc thu nhỏ hóa và tối ưu hiệu suất hoạt động của các loại cảm biến, pin đã trở thành một yếu tố then chốt, mang tính quyết định trong quá trình tích hợp chúng vào các thiết bị điện tử có kích thước nhỏ gọn. Không chỉ góp phần nâng cao độ chính xác trong việc đo lường các thông số sinh lý quan trọng, nó còn giúp giảm thiểu kích thước thiết bị, tăng tính di động và mở rộng khả năng ứng dụng trong nhiều lĩnh vực, đặc biệt nổi bật trong y học cá thể hóa.

Bên cạnh đó, sự phát triển vượt bậc của trí tuệ nhân tạo (AI) đóng vai trò như một đòn bẩy quan trọng trong việc nâng cao hiệu quả khai thác dữ liệu cảm biến. AI không chỉ tối ưu hóa quy trình xử lý và phân tích dữ liệu mà còn thúc đẩy khả năng nhận diện mẫu, phân cụm, dự đoán và đưa ra quyết định mang tính hỗ trợ lâm sàng. Trong lĩnh vực chẩn đoán hội chứng ngưng thở khi ngủ, sự phối hợp giữa cảm biến và thuật toán học máy không chỉ đảm bảo độ chính xác cao trong thu thập dữ liệu mà còn mở ra khả năng phân tích chuyên sâu về các yếu tố sinh lý, phục vụ quá trình đánh giá y tế toàn diện.

Trên cơ sở đó, nghiên cứu này tập trung phát triển một thiết bị theo dõi tình trạng bệnh nhân trong khi ngủ, sử dụng cảm biến gia tốc kết hợp với thuật toán học máy nhằm phân tích tư thế ngủ và phát hiện các dấu hiệu liên quan đến hội chứng ngưng thở khi ngủ. Hệ thống được tiến hành nghiên cứu, triển khai trên ứng dụng di động trước rồi được tích hợp trên chip, hướng đến khả năng giám sát liên tục, thu thập dữ liệu thời gian thực và đưa ra đánh giá có độ tin cậy cao, góp phần cải thiện hiệu quả trong phát hiện sớm và hỗ trợ điều trị bệnh lý hô hấp liên quan

đến giấc ngủ.

**Từ khóa:** cảm biến gia tốc, học máy, ngưng thở khi ngủ

# Mục lục

# Danh sách hình vẽ

# **Danh sách bảng**

# Đặt vấn đề

Ngưng thở tắc nghẽn khi ngủ (Obstructive Sleep Apnoea – OSA) là một rối loạn hô hấp phổi biến trong giấc ngủ, được đặc trưng bởi các đợt ngưng thở hoặc giảm thông khí tắc nghẽn lặp đi lặp lại trong lúc ngủ, dẫn đến việc gián đoạn giấc ngủ do vi thức giấc và giảm oxy trong máu. Tỷ lệ hiện mắc OSA tại Việt Nam ước tính khoảng 8,5% [[nguoimacOSA\\_VN](#)]. OSA hiện được công nhận là một yếu tố có nguy cơ độc lập đối với nhiều bệnh lý liên quan đến tim mạch, đặc biệt là tăng huyết áp. Ngoài ra, hội chứng này còn có mối liên hệ đáng kể với các nguy cơ như tai nạn giao thông, tai nạn lao động dẫn đến làm gia tăng gánh nặng kinh tế xã hội [[osa\\_bike](#)][[Marin2005](#)][[drive\\_osa](#)]. Đáng chú ý, tình trạng ngưng thở khi ngủ kéo dài và không được phát hiện, điều trị có thể ảnh hưởng nghiêm trọng đến sức khỏe gây ra rối loạn nhịp tim và một trong những nguyên nhân gây đột tử [[sumarry\\_osa](#)]. Theo PSG.TS Nguyễn Thy Khuê, Hội Y học Giấc ngủ Việt Nam, ngưng thở khi ngủ còn là một trong những dấu hiệu rõ ràng của bệnh đái tháo đường, bệnh thận [[bsThyKhue](#)]. OSA được phát hiện ở hơn 20% người bệnh đái tháo đường và làm trầm trọng thêm các rối loạn chuyển hóa, đặc biệt là đái tháo đường type 2. Trong một nghiên cứu tiêu biểu tại Việt Nam, GS.TS. Dương Quý Sỹ và cộng sự đã khảo sát 524 trẻ em mắc rối loạn tăng động giảm chú ý (Attention Deficit Hyperactivity Disorder - ADHD) tại Bệnh viện Nhi Trung ương Việt Nam. Kết quả cho thấy tỷ lệ mắc (OSA) ở nhóm này là 23.3%, trong đó chủ yếu ở mức độ trung bình đến nặng [[ThaySUCHildren](#)]. Nghiên cứu cũng đồng thời xác định mối tương quan đáng kể giữa mức độ nghiêm trọng của OSA và các triệu chứng mất tập trung, tăng động, rối loạn hành vi, lo âu và trầm cảm. Phát hiện nhấn mạnh sự cần thiết của việc sàng lọc OSA trong quá trình điều trị toàn diện ADHD ở trẻ em.

Một dạng đặc biệt của OSA được ghi nhận là ngưng thở khi ngủ do tư thế (Positional Obstructive Sleep Apnea - pOSA). Bệnh nhân được chẩn đoán mắc pOSA có chỉ

số AHI lớn hơn 5, và giá trị AHI ở tư thế ngửa cao gấp ít nhất hai lần so với nằm ở tư thế khác [heinzer2018]. Các nghiên cứu gần đây chỉ ra rằng tỷ lệ mắc pOSA lên tới 50% bệnh nhân OSA [sabil2020]. Điều này cho thấy tư thế ngửa có ảnh hưởng sinh lý rõ rệt đến sự sụp đổ đường thở trên, đặc biệt ở tư thế nằm ngửa. Lúc nằm ngửa, trọng lực làm xẹp các cơ vùng họng dẫn đến làm hẹp khoang khí.

Việc chẩn đoán hội chứng ngưng thở tắc nghẽn khi ngủ (OSA) hiện nay chủ yếu được thực hiện thông qua hai phương pháp: đa ký giấc ngủ (Polysomnography – PSG) và thiết bị kiểm tra giấc ngủ tại nhà (Home Sleep Test – HST). Trong đó, PSG được xem là tiêu chuẩn vàng trong việc đánh giá OSA. DO phương pháp này cho phép thu thập đồng thời nhiều thông số sinh lý quan trọng bao gồm: luồng khí hô hấp qua mũi và/hoặc miệng, cử động thành ngực và bụng, tiếng ngáy, điện não đồ (Electroencephalography – EEG), điện tâm đồ (Electrocardiography – ECG), điện cơ đồ (Electromyography – EMG), và độ bão hòa oxy trong máu (SpO). Quá trình do PSG phải được thực hiện trong môi trường có kiểm soát tại các cơ sở y tế chuyên khoa, dưới sự giám sát trực tiếp của bác sĩ chuyên ngành giấc ngủ hoặc kĩ thuật viên có chuyên môn[psg\_paper][kushida2005psg].

Mặc dù PSG vẫn giữ vai trò là phương pháp tham chiếu trong chẩn đoán và theo dõi chất lượng giấc ngủ cũng như các rối loạn liên quan, nhưng việc triển khai kỹ thuật này thường đòi hỏi chi phí cao, trang thiết bị chuyên dụng và điều kiện thực hiện tại các cơ sở y tế chuyên khoa. Một thách thức khác của PSG là người bệnh thường cảm thấy bất tiện và cảm giác khó chịu do mang nhiều cảm biến gắn trên cơ thể trong suốt đêm, dẫn đến nguy cơ gián đoạn hoặc sai lệnh dữ liệu trong quá trình ghi nhận. Vì vậy, các hệ thống HST đang ngày càng thu hút sự quan tâm từ cộng đồng khoa học toàn cầu[e3hst][hstSurvey][hst\_paper]. Các thiết bị HST hiện đại dùng các cảm biến không xâm lấn nhằm ghi nhận và phân tích một số tín hiệu sinh lý cơ bản như luồng khí hô hấp, tư thế ngủ, áp suất mũi, độ bão hòa oxy và nhịp tim. Việc cải thiện chất lượng, kéo dài thời lượng sử dụng và tăng độ chính xác và cải thiện mức độ thoái mái vẫn là những thách thức lớn đối với giới nghiên cứu và các nhà sản xuất thiết bị y tế. Tuy nhiên, với những tiến bộ công nghệ đang diễn ra nhanh chóng, HST có tiềm năng trở thành một công cụ chẩn đoán quan trọng và được ứng dụng rộng rãi trong lâm sàng. Điều này không chỉ mang lại sự thuận tiện và chấp nhận cao hơn từ phía người bệnh, mà còn góp phần làm giảm gánh nặng chi phí và áp lực cho hệ thống chăm sóc sức khỏe.

Trong những năm gần đây, nhiều nhóm nghiên cứu đang chú trọng nghiên cứu, phát

triển hệ thống HST nhằm mục đích thay thế hoặc hỗ trợ cho đa ký giấc ngủ. Năm 2011, Collop và cộng sự đã phát triển một hệ thống phân loại SCOPER (Sleep, Cardiovascular, Oximetry, Position, Effort, and Respiration) để đánh giá các tín hiệu sinh lý thu nhận trong việc chẩn đoán OSA [**hst\_6p\_paper**]. Tác giả Morillo D và cộng sự đề xuất một phương pháp sàng lọc ngưng thở tắc nghẽn khi ngủ dựa trên cảm biến gia tốc gắn tại vị trí hõm úc, cho phép trích xuất các tín hiệu hô hấp, tim mạch và tiếng ngáy bằng kỹ thuật xử lý tín hiệu số, từ đó chứng minh tính khả thi của thiết bị di động đơn giản và chi phí thấp trong hỗ trợ chẩn đoán hội chứng ngưng thở – giảm thở khi ngủ [**morillo2010accelerometer**]. Trong nhóm các thiết bị đeo, A.H. Yüzer và cộng sự đã phát triển thiết bị đeo tay sử dụng cảm biến gia tốc ADXL345 để phát hiện và phát tín hiệu rung khi cảnh báo. [**hst\_wear\_paper**]. Tương tự, nhóm Yunyoung Nam và cộng sự cũng đã tích hợp hệ thống thu thập, phân tích sử dụng một cảm biến gia tốc ba trục và một cảm biến áp suất để giám sát chất lượng giấc ngủ tư thế ngủ, trạng thái ngủ, giai đoạn ngủ (REM và chu kỳ giai đoạn ngủ không REM) [**hst\_pressure\_paper**]. Tại Việt Nam, nhóm nghiên cứu của Giáo sư Lê Tiến Thường, trường đại học Bách Khoa TP Hồ Chí Minh đã sử dụng cảm biến gia tốc MPU6050 cùng với vi xử lý ESP32 nhằm ghi nhận hơi thở và nhịp tim của bệnh nhân OSA thông qua rung động, và dòng chảy của động mạch và tĩnh mạch ở cổ [**thuong\_wear\_paper**]. Gần đây, Domingues và cộng sự (2024) xây dựng một mô hình mạng nơ-ron nhân tạo dựa trên dữ liệu từ máy đo SPO2, cảm biến gia tốc và ghi âm tiếng ngáy của hệ thống Biologix, nhằm dự đoán chính xác trạng thái ngủ. Kết quả cho thấy mô hình này có khả năng cao độ chính xác trong chẩn đoán ngưng thở khi ngủ tại nhà, tiệm cận với tiêu chuẩn của đa ký giấc ngủ truyền thống [**domingues2024sleep**]. Một hướng nghiên cứu khác là cảm biến đặt dưới nệm giường. Tác giả Andrei Boiko và cộng sự đánh giá hệ thống phát sử dụng cảm biến gia tốc đặt dưới đệm giường để ghi dao động do cử động ngực khi thở. Kết quả cho thấy thuật toán phát hiện ngưng thở đạt độ chính xác, độ đặc hiệu và độ nhạy lần lượt là 94.6%, 95.3% và 93.7% [**Boiko2023**].

Nhìn chung, việc chẩn đoán OSA hay pOSA đòi hỏi hệ thống cần có khả năng thu thập liên tục, và đưa ra quyết định chính xác dựa trên những dữ liệu đó. Do đó, việc xây dựng các mô hình học máy để phân loại AHI, hay các tư thế ngủ đang là 1 hướng nghiên cứu phổ biến hợp hợp cả hai lĩnh vực y tế giấc ngủ và công nghệ dữ liệu. Các nghiên cứu tiêu biểu sử dụng mô hình như Decision Tree, Random Forest, SVM và các mô hình Deep Learning cho độ chính xác từ

Khóa luận văn có mục đích nghiên cứu, ứng dụng cảm biến gia tốc kết hợp với các bộ lọc tín hiệu, trí tuệ nhân tạo, ứng dụng phần mềm để tạo ra một sản phẩm có độ chính xác, tính ứng dụng cao trong cuộc sống. Khóa luận đưa ra cái nhìn tổng quan về chứng ngưng thở khi ngủ và thiết bị chẩn đoán sử dụng cảm biến gia tốc. Kết quả có tiềm năng ứng dụng thực tế phục vụ cho mục đích chuẩn đoán, đánh giá và phân tích về chứng ngưng thở khi ngủ. Khóa luận này trình bày thiết kế, chế tạo và thử nghiệm một thiết bị đeo được tích hợp cảm biến gia tốc 3 trục cho phép tiến hành thực nghiệm chế tạo và thiết lập hệ thống đo đặc cho cảm biến đã đề xuất với những mục tiêu cụ thể: 01) Tăng độ chính xác của hệ thống, 02) Thu nhỏ thiết kế mạch, 03) Tăng trường khả năng lưu trữ dữ liệu và 04) Hỗ trợ người sử dụng thông qua phần mềm ứng dụng. Khóa luận được thực hiện thông qua 01) tham khảo tài liệu, 02) Phương pháp nghiên cứu lý thuyết, 03) phương pháp thực nghiệm. Nội dung nghiên cứu được trình bày trong 3 chương như sau:

Chương 1: Tổng quan về OSA và các thiết bị theo dõi.

Chương 2: Quá trình lựa chọn và xây dựng thiết bị.

Chương 3: Kết quả thực nghiệm, đánh giá và kế hoạch nghiên cứu trong thời gian tới.

# **Chương 1**

## **Tổng quan về OSA 3 và các thiết bị theo dõi**

### **1.1. Giới thiệu về chứng ngưng thở trong lúc ngủ**

OSA[osa\_summary][ThaySYOSA] là tình trạng xảy ra khi đường hô hấp trên bị tắc nghẽn một phần hoặc hoàn toàn trong ít nhất 10 giây, mặc dù cơ thể vẫn nỗ lực hô hấp thông qua chuyển động của ngực và bụng. Các triệu chứng thường gặp của OSA bao gồm buồn ngủ quá mức vào ban ngày, mệt mỏi, ngáy to, thức giấc ngắn lặp đi lặp lại (vi thức tỉnh), và đau đầu khi thức dậy vào buổi sáng. OSA làm độ bão hòa oxy trong máu giảm xuống khi đó các cơ sẽ cố gắng để chống lại việc này khiến người bị OSA sẽ có thể thức dậy. Các mức độ của hội chứng ngưng thở khi ngủ do tắc nghẽn (OSA) được đánh giá dựa trên chỉ số ngưng thở giảm thở (Apnea–Hypopnea Index - AHI) bằng cách chia tổng số lần ngưng thở và hẹp thở cho tổng số giờ đã ngủ, với mỗi sự kiện phải kéo dài ít nhất 10 giây ??.

1) Ngưng thở tắc nghẽn (Obstructive Apnea): Là tình trạng ngưng thở gần như hoàn toàn (>90%) trong thời gian ít nhất 10 giây, mặc dù vẫn có nỗ lực hô hấp từ lồng ngực và bụng. 2) Giảm hô hấp (Hypopnea): Là sự giảm luồng khí hô hấp > 30%, đi kèm với giảm độ bão hòa oxy trong máu >3% hoặc gây thức giấc. Tỷ lệ ngưng thở tắc nghẽn là từ 2% đến 9% ở người lớn. Ngưng thở tắc nghẽn khi ngủ có thể tăng gấp 4 lần ở nam giới và gấp 7 lần hơn ở những người béo phì (ví dụ chỉ số khối cơ thể (Body mass Index

Bảng 1.1: Phân loại mức độ ngưng thở khi ngủ theo chỉ số AHI

AHI	Cấp độ
<5	Không mắc
5 đến 10	Nhẹ
15 đến 30	Trung bình
>30	Nặng

- BMI) > 30). OSA nặng (AHI > 30/h) làm tăng nguy cơ tử vong ở nam giới trung niên.

Ngưng thở khi ngủ do tắc nghẽn thường xảy ra ở người lớn tuổi và những người thừa cân béo phì. Yếu tố gây ra có thể liên quan đến cấu trúc hoặc phi cấu trúc, bao gồm cả yếu tố di truyền. Nguyên nhân chủ yếu là do diện tích vòng họng hoặc khoang mũi bị thu hẹp như viêm xoang, các khối u, bệnh phì đại tuyến lưỡi, amydal, phì đại tuyến mõ (đặc biệt ở trẻ em), béo phì hoặc đến từ các bệnh lý: tiểu đường, huyết áp cao, các bệnh tim mạch v.v[**wright1997health**]. Ngoài ra, có thể đến từ thói quen không lành mạnh của con người như là sử dụng các chất kích thích, hút thuốc, ngáy khi ngủ[**reason\_osa**][**reasonOsa**]. Bên cạnh đó, các yếu tố không giải phẫu như hoạt động kém của cơ giän họng, ngưỡng thức giấc thấp và sự điều hòa hô hấp không ổn định cũng góp phần quan trọng vào cơ chế bệnh sinh. Sự tương tác giữa các yếu tố này tạo nên tính đa dạng trong biểu hiện và mức độ nặng của OSA.

Phần lớn bệnh nhân mắc hội chứng ngưng thở khi ngủ tắc nghẽn (OSA) không tự nhận thức được các rối loạn hô hấp xảy ra trong lúc ngủ. Điều này đặc biệt đúng với những người sống hoặc ngủ một mình, do thiếu sự quan sát từ bên ngoài. Đặc biệt OSA xuất hiện lớn hơn 80% ở các bệnh liên quan đến béo phì: tiểu đường, thận, mõ máu v.v[**wright1997health**]. Hội chứng ngưng thở khi ngủ tắc nghẽn (OSA) ảnh hưởng nghiêm trọng đến chất lượng cuộc sống của người bệnh. Các hoạt động thường ngày trở nên gián đoạn hoặc khó thực hiện hơn. Hậu quả của OSA bao gồm suy giảm trí nhớ, giảm tinh táo, dễ cáu gắt, trầm cảm, đau đầu và giảm khả năng tập trung[**flemons1997quality**]. Những tác động này dẫn đến hiệu suất làm việc giảm, rối loạn các mối quan hệ xã hội và gia tăng nguy cơ tai nạn giao thông. Một nghiên cứu khác cũng bởi Mooke và cộng sự[**mooke1996sleep**] ở nam giới cho thấy 37% bệnh nhân bệnh mạch vành (Coronary Artery Disease - CAD) có AHI > 10. Young và cộng sự (1997) phát hiện

rằng chỉ số AHI tăng tỉ lệ thuận với huyết áp tâm thu và tâm trương ( $p=0.003$  và  $p=0.01$ )[young1997population].

Trong điều kiện hiện tại, đa số bệnh nhân nghi ngờ có hội chứng tắc nghẽn ngưng thở khi ngủ được khám bởi bs chuyên khoa Tai Mũi Họng và bác sĩ chuyên gia về ngủ ngáy. Khám tổng quát hỏi bệnh sử liên quan với thang điểm ngủ ngáy (ví dụ: Epworth Sleepiness Scale, STOP BANG đã được dịch sang tiếng Việt và được chấp thuận tại Việt Nam như một phương án sàng lọc bệnh nhân OSA) và khám nội soi Tai Mũi Họng để tìm nguyên nhân. Vì đa số các trường hợp ngáy, ngưng thở khi ngủ là do tắc nghẽn ngoại biên, nguyên nhân từ Mũi – Họng – Màn hầu , VA và amidan, và những bất thường về hàm mặt khác. Việc đánh giá ngưng thở lúc ngủ bắt đầu bằng đánh giá giấc ngủ toàn diện, bao gồm khai thác bệnh sử về các triệu chứng hoặc dấu chứng ban đầu sau đó là đánh giá tổng thể qua PSG[diagnosis\_osa][medical2006polysomnography]. Phương pháp do dùng đa ký giấc ngủ (Polysomnography) với sự giám sát của các bác sĩ chuyên môn được coi là tiêu chuẩn vàng trong chẩn đoán chứng ngưng thở khi ngủ. Polysomnography là một phương pháp ghi đa kênh liên tục trong suốt một đêm, bao gồm nhiều thông số sinh lý nhằm đánh giá toàn diện hoạt động hô hấp và thần kinh khi ngủ. Các thành phần chính trong một xét nghiệm polysomnography bao gồm: điện não đồ (EEG) để ghi lại hoạt động điện của não; điện cơ ký (EMG) nhằm đo trương lực cơ, đặc biệt là ở cẳng và chân; điện động mắt (EOG) để theo dõi chuyển động của nhãn cầu, giúp xác định các giai đoạn của giấc ngủ; và điện tâm đồ (ECG) để theo dõi hoạt động điện của tim. Bên cạnh đó, quá trình đo cũng bao gồm theo dõi độ bão hòa oxy trong máu (SpO2), đo lưu lượng khí thở qua mũi và miệng, đánh giá nỗ lực hô hấp thông qua chuyển động của ngực và bụng, đo áp lực khí thở qua mũi, và ghi nhận cường độ tiếng ngáy. Tập hợp các thông số này cho phép bác sĩ chẩn đoán chính xác các rối loạn hô hấp khi ngủ, đặc biệt là hội chứng ngưng thở khi ngủ tắc nghẽn (OSA).

Một trong những hạn chế của phương pháp đánh giá sử dụng (PSG) là sự bất tiện, chi phí cao và khả năng phổ biến thấp, nhất là đối với phần lớn người bệnh có thu nhập thấp. Việc yêu cầu bệnh nhân phải lưu trú qua đêm tại cơ sở y tế, cùng với việc gắn nhiều thiết bị theo dõi sinh lý lên cơ thể, không chỉ gây cảm giác khó chịu mà còn tiềm ẩn nguy cơ ảnh hưởng đến chất lượng và tính chính xác của dữ liệu thu thập được. Chính những bất cập này đã thúc đẩy sự phát triển của các thiết bị theo dõi giấc ngủ ngoài trung tâm (Out-of-Center devices) hay còn gọi là thiết bị kiểm tra giấc ngủ tại nhà (Home Sleep

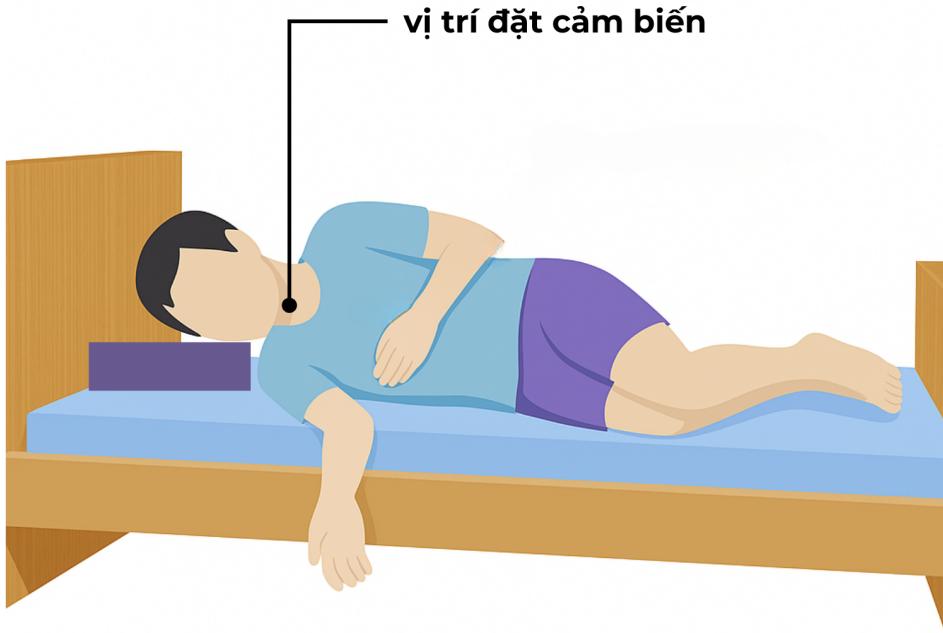
Test – HST). Những thiết bị này thường được thiết kế với số lượng cảm biến tối giản hơn so với PSG truyền thống, đồng thời tích hợp các thuật toán phân tích tự động – được xử lý trực tiếp trên thiết bị hoặc thông qua phần mềm chuyên dụng – nhằm hỗ trợ chẩn đoán ngưng thở khi ngủ do tắc nghẽn (OSA) một cách thuận tiện và tiết kiệm hơn. Những thông số SCOPERA được coi là cơ sở để xây dựng thiết bị HST trong đó giấc ngủ (Sleep - S), tim mạch (Cardiovascular - C), oxi trong máu (Oximetry - O), cỗ gắng thở (Effort - E), luồng không khí lưu thông (Respiratory - R), âm thở (Audio - A). Nghiên cứu[svmHSt2017] chứng minh rằng tín hiệu chuyển động ngực (Thoracic movement signal - THO) và bụng (Abdominal movement signal - ABD), thu từ các dải piezoelectric đeo được, có thể được sử dụng hiệu quả để phân loại các dạng rối loạn thở khi ngủ thông qua mô hình thuật toán SVM. Kết quả cho thấy khi kết hợp cả hai tín hiệu, độ chính xác phân loại đạt trung bình 81.8%, khẳng định tiềm năng ứng dụng của phương pháp này trong sàng lọc và theo dõi OSA tại nhà hoặc trong lâm sàng. Theo tìm hiểu của tác giả, thiết bị HST vẫn có cấu hình khá phức tạp với nhiều cảm biến và thao thác cũng chưa hoàn toàn đơn giản, thêm vào đó giá trung bình của thiết bị vào khoảng 2300 USD làm hạn chế khả năng tiếp cận của người sử dụng, đặc biệt là những người có thu nhập trung bình hoặc thấp [hst\_review]. Do đó, những nghiên cứu gần đây tập trung theo hướng phát triển thiết bị với ít điện cực hơn để tối giản phần cứng của hệ thống cũng như tích hợp thêm các mô hình học máy hoặc trí tuệ nhân tạo làm tăng tính chính xác và độ lặp lại. Trong luận văn này, tác giả lựa chọn nghiên cứu ban đầu sử dụng cảm biến gia tốc để phát hiện các tư thế ngủ có liên quan tới chứng ngưng thở trong lúc ngủ kết hợp việc xây dựng mô hình học máy hướng tới phát triển một thiết bị IoT có thể dự đoán mức độ nghiêm trọng của chứng ngưng thở trong lúc ngủ thông qua chỉ số AHI.

## **1.2. Ứng dụng cảm biến gia tốc trong đánh giá tư thế ngủ của người mắc OSA tại nhà**

Đánh giá tư thế ngủ là một thành phần quan trọng trong nghiên cứu giấc ngủ, đặc biệt trong chẩn đoán và theo dõi các rối loạn hô hấp liên quan đến giấc ngủ như hội chứng ngưng thở khi ngủ do tắc nghẽn (OSA). Nhiều phương pháp kỹ thuật đã được phát triển nhằm ghi nhận và phân tích tư thế ngủ một cách chính xác. Các hệ thống ghi hình sử dụng camera hồng ngoại có khả năng thu thập toàn bộ quá trình ngủ trong điều kiện

ánh sáng thấp, trong khi đó, các cảm biến gia tốc ba trục (triaxial accelerometers) cho phép nhận diện sự thay đổi tư thế dựa trên dao động và gia tốc của cơ thể. Bên cạnh đó, các thiết bị tích hợp cảm biến áp suất, chẳng hạn như đệm cảm biến, cung cấp thông tin về sự phân bố trọng lực và áp lực tiếp xúc, từ đó suy luận tư thế ngủ một cách gián tiếp nhưng hiệu quả. Những hệ thống tiên tiến hơn còn kết hợp đa cảm biến và tích hợp nhiều phương pháp đồng thời, nhằm nâng cao độ tin cậy, tính định lượng và khả năng ứng dụng trong cả môi trường lâm sàng lẫn tại nhà. Theo S. Akbarian và đồng các tác giả đã đề cập đến phương pháp giám sát tư thế ngủ bằng máy ảnh hồng ngoại kết hợp với công nghệ học sâu (Deep learning) [**Akbarian\_osa**] có kết quả tốt nhưng có khó khăn để xác định các góc các nhau của khuôn mặt. Còn theo A. Channa, M. Yousuf và N. Popescu đã sử dụng cảm biến áp suất được gắn dưới đệm để theo dõi tư thế ngủ với 2048 điểm cảm biến [**Channa\_osa**]. Trong đó cảm biến gia tốc 3 trục được đánh giá là phương pháp hiệu quả, tiết kiệm chi phí và độ chính xác cao. Jeng PY và đồng nghiệp đã thực hiện phát triển chế tạo thiết bị đeo tay sử dụng cảm biến gia tốc kết hợp với thiết bị ở ngực để lấy nhãn cho dữ liệu. Các phương pháp học máy truyền thống được sử dụng với độ chính xác đánh giá 4 tư thế khi ngủ trên 85% [**Jeng\_osa**]. Cách đặt và vị trí đặt cảm biến ảnh hưởng rất lớn đến chất lượng, độ chính xác của phép đo. Theo đó vị trí xương ức cổ được xem là vị trí có thể thu được tín hiệu chính xác để đặt đơn cảm biến [**Zhang\_osa**]. Việc sử dụng nhiều cảm biến ở những vị trí khác như cổ tay, trán, ... sẽ có thêm nhiều dữ liệu hữu ích để phân tích, đánh giá. Ngoài ra, hiện nay với sự phát triển vượt bậc của điện thoại di động, việc tận dụng cảm biến gia tốc ở ngay trên chính chiếc điện thoại cũng là giải pháp hữu hiệu. Nhóm tác giả trong [**Ferrer\_osa**] đã báo cáo nghiên cứu đánh giá tư thế ngủ của bệnh nhân sử dụng thiết bị di động đeo ở xương ức kết hợp với phần mềm trên nền tảng Android để thu thập lại dữ liệu kể cả khi tắt màn hình. Trong một nghiên cứu tiêu biểu, Natale và cộng sự đã khai thác các cảm biến tích hợp sẵn trên điện thoại iPhone để ước lượng các thông số liên quan đến chất lượng giấc ngủ, bao gồm tổng thời gian ngủ (Total Sleep Time – TST), độ trễ vào giấc (Sleep Onset Latency – SOL) và hiệu quả giấc ngủ (Sleep Efficiency – SE). Phương pháp tiếp cận này cho thấy tiềm năng trong việc sử dụng thiết bị di động như một công cụ theo dõi giấc ngủ tiện lợi và dễ tiếp cận, đặc biệt trong các nghiên cứu cộng đồng và ứng dụng tại nhà [**Natale\_osa**]. Đặc điểm của sử dụng tích hợp cảm biến gia tốc trên điện thoại là rất tiện lợi, sử dụng trực tiếp mà không cần phát triển phần cứng. Tuy nhiên, việc tiếp xúc điện thoại trực tiếp với cơ thể trực tiếp trong thời gian lâu cũng có gây những ảnh hưởng

nhất định đến người dùng.

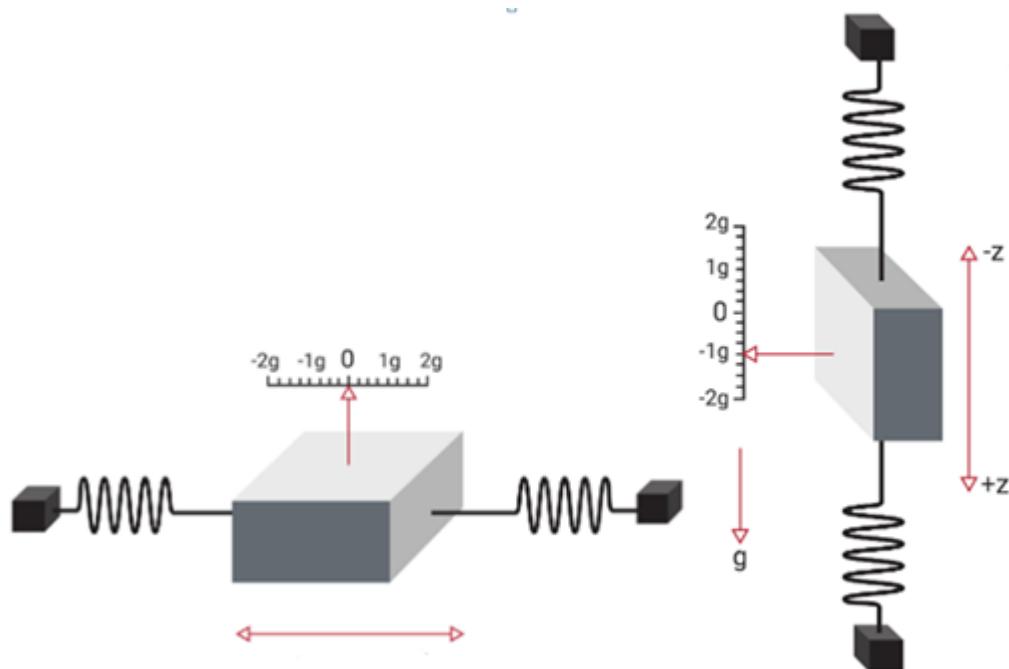


Hình 1.1: Vị trí tối ưu để đặt cảm biến gia tốc

Trong khuôn khổ luận văn, tác giả đề xuất thiết kế một thiết bị đeo tiếp xúc sử dụng cảm biến gia tốc được đặt tại vị trí xương ức cổ nhằm theo dõi và phân tích tư thế ngủ của người dùng. Vị trí này được lựa chọn không chỉ do tính ổn định trong quá trình ngủ mà còn thuận lợi để tích hợp thêm các cảm biến khác như cảm biến âm thanh và cảm biến nhiệt độ – phục vụ cho các mục tiêu nghiên cứu mở rộng của nhóm. Tín hiệu từ cảm biến gia tốc sẽ được thu thập dưới dạng ba trục không gian ( $x$ ,  $y$ ,  $z$ ), phản ánh chuyển động và hướng trọng lực tương ứng với tư thế cơ thể trong suốt thời gian ngủ. Sau quá trình thu thập, dữ liệu gia tốc sẽ được xử lý sơ cấp bao gồm hiệu chỉnh, lọc nhiễu, và chuẩn hóa nhằm đảm bảo tính chính xác và đồng nhất giữa các mẫu đo. Tiếp theo, các đặc trưng định lượng (features) trong miền thời gian sẽ được trích xuất để phục vụ cho bài toán phân loại tư thế ngủ (ngửa, nghiêng trái, nghiêng phải, sấp). Các đặc trưng này cùng với dữ liệu gốc sẽ được lưu trữ trong hệ thống để phục vụ cho các bước phân tích tiếp theo, bao gồm huấn luyện mô hình học máy hoặc tích hợp với các chỉ số sinh lý khác trong đánh giá rối loạn giấc ngủ, đặc biệt là hội chứng ngưng thở khi ngủ (OSA) hình ??.

### 1.3. Giới thiệu về cảm biến gia tốc nhiều bậc tự do

Cảm biến gia tốc (accelerometer) là thiết bị được sử dụng phổ biến để đo và phân tích gia tốc của một vật thể. Nhờ khả năng phát hiện sự thay đổi về chuyển động, cảm biến này được ứng dụng rộng rãi trong nhiều lĩnh vực như phát hiện rơi tự do, va chạm, dịch chuyển, rung động, và xoay. Nguyên lý hoạt động chính của cảm biến gia tốc dựa trên định luật II Newton ( $F = ma$ ), theo đó khi một lực tác động lên một khối lượng, nó sẽ sinh ra gia tốc. Trong cấu trúc cảm biến, sự thay đổi này được ghi nhận thông qua việc chuyển đổi qua lại giữa năng lượng cơ học (sự dịch chuyển của khối lượng) và năng lượng điện (qua sự thay đổi điện tích, điện dung hoặc điện áp). Chính khả năng chuyển đổi năng lượng này giúp cảm biến gia tốc hoạt động hiệu quả trong việc giám sát và ghi nhận các trạng thái động học của vật thể.



Hình 1.2: Nguyên lý cơ bản của cảm biến gia tốc

Như minh họa trong Hình ??, khi cảm biến gia tốc chịu tác động từ một chuyển động, khối gia trọng (proof mass) sẽ dịch chuyển, làm lò xo kết nối bị biến dạng. Sự biến dạng này tạo ra một lực đàn hồi theo định luật Hooke, tỷ lệ thuận với độ giãn của lò xo. Áp dụng định luật II Newton, ta có mối quan hệ giữa lực, khối lượng và gia tốc như sau:

$$F = m \cdot a \Rightarrow a = \frac{k \cdot \Delta l}{m} \quad (1.1)$$

Trong đó:

- $F$  là lực đàn hồi tác dụng lên khối gia trọng (N)
- $m$  là khối lượng của khối gia trọng (kg)
- $k$  là hệ số đàn hồi của lò xo (N/m)
- $\Delta l$  là độ biến dạng (thay đổi chiều dài) của lò xo (m)

Phương trình trên cho thấy gia tốc có thể được tính toán gián tiếp thông qua độ biến dạng của lò xo, từ đó cho phép cảm biến gia tốc chuyển đổi dao động cơ học thành tín hiệu điện phục vụ cho việc đo đặc và phân tích chuyển động. Trong hệ tọa độ của cảm biến gia tốc ba trục, trục z thường nằm theo phương vuông góc với mặt phẳng ngang và sẽ chịu thêm tác dụng của trọng lực. Do đó, ở trạng thái cân bằng (khi thiết bị đứng yên và không có chuyển động nào khác), giá trị gia tốc đo được tại trục z sẽ xấp xỉ bằng gia tốc trọng trường  $g$  (khoảng  $9.81 \text{ m/s}^2$ ). Đặc điểm này có thể được khai thác trong việc hiệu chuẩn cảm biến cũng như xác định tư thế không gian tương đối của thiết bị.

Trong khuôn khổ luận văn này, tác giả tập trung vào việc nghiên cứu và ứng dụng cảm biến gia tốc được chế tạo dựa trên công nghệ vi cơ điện tử (Micro-Electro-Mechanical Systems – MEMS). Đây là một công nghệ tiên tiến cho phép tích hợp các cấu trúc cơ học và linh kiện điện tử ở kích thước vi mô (dưới 10 micromet) trên cùng một chip bán dẫn. Một trong những ưu điểm vượt trội của cảm biến gia tốc MEMS là khả năng gắn trực tiếp lên bo mạch in (PCB), giúp tiết kiệm không gian, giảm chi phí sản xuất và tối ưu hóa thiết kế hệ thống nhúng – đặc biệt phù hợp với các ứng dụng trong thiết bị đeo, điện thoại di động, và hệ thống theo dõi sức khỏe cá nhân.

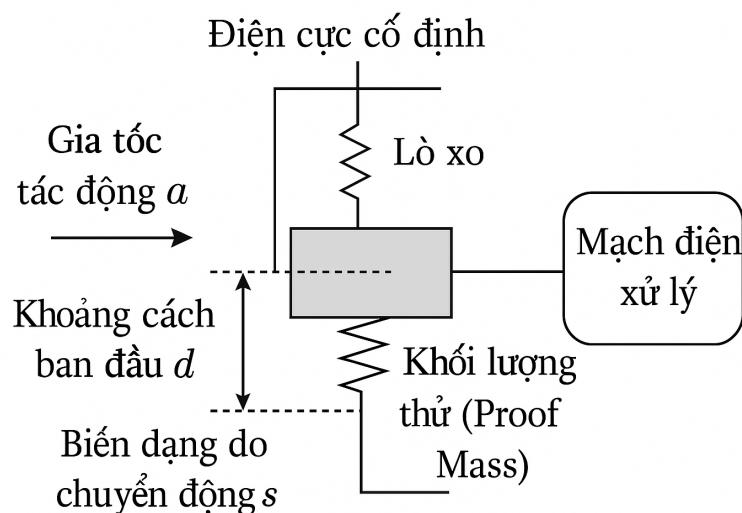
Hiện nay, có ba loại cảm biến gia tốc MEMS phổ biến, được phân loại dựa trên nguyên lý hoạt động của chúng [Acce][cambien]:

- **Cảm biến gia tốc dựa trên hiệu ứng điện dung (Capacitive accelerometers):**  
Đây là loại cảm biến được sử dụng rộng rãi nhất trong các thiết bị điện tử tiêu dùng như điện thoại thông minh và thiết bị đeo. Nguyên lý hoạt động dựa trên sự thay

đổi điện dung giữa các bản cực khi khói gia trọng dịch chuyển dưới tác dụng của gia tốc. Sự thay đổi điện dung này sẽ được chuyển đổi thành tín hiệu điện tương ứng với mức gia tốc.

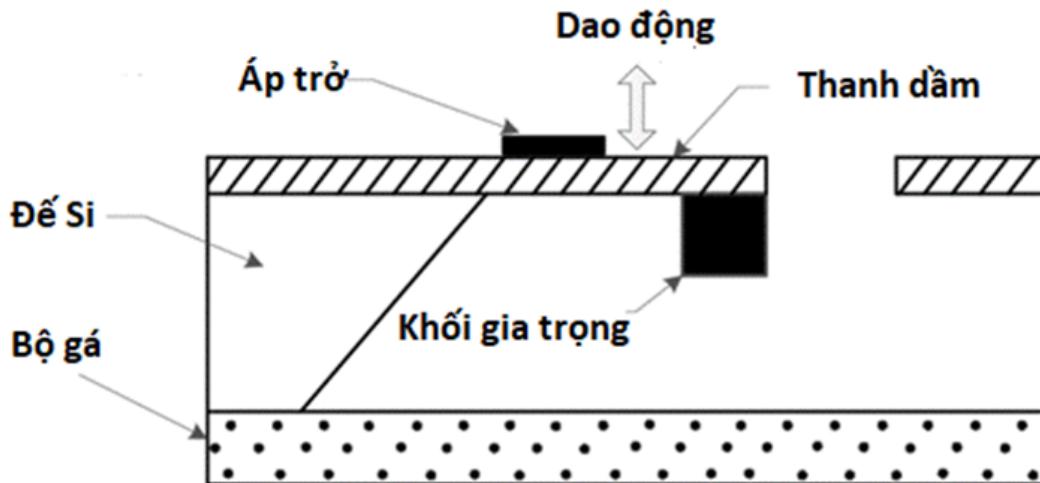
- **Cảm biến gia tốc dựa trên hiệu ứng áp điện trở (Piezoresistive accelerometers):** Trong loại cảm biến này, lực hoặc ứng suất cơ học tác động lên cảm biến sẽ làm thay đổi điện trở của vật liệu bán dẫn bên trong. Hiện tượng này – được gọi là hiệu ứng áp điện trở – có đặc tính tuyến tính, trong đó độ biến đổi của điện trở tỷ lệ thuận với lực tác động. Cảm biến loại này thường được ứng dụng trong môi trường có điều kiện khắc nghiệt, do khả năng chịu nhiệt và độ bền cao.
- **Cảm biến gia tốc dựa trên hiệu ứng áp điện (Piezoelectric accelerometers):** Loại cảm biến này khai thác hiện tượng áp điện, trong đó lực cơ học tác động lên các tinh thể áp điện sẽ tạo ra điện tích. Hiệu ứng áp điện có tính tuyến tính, và lượng điện tích sinh ra tỷ lệ thuận với độ lớn của lực. Cảm biến này phù hợp trong các ứng dụng đo rung động hoặc gia tốc có tần số cao.

Mỗi loại cảm biến kể trên đều có những ưu điểm và hạn chế riêng, và việc lựa chọn loại cảm biến phù hợp phụ thuộc vào yêu cầu cụ thể của từng ứng dụng, bao gồm độ chính xác, dải đo, mức tiêu thụ điện năng và điều kiện môi trường hoạt động.



Hình 1.3: Cấu trúc cảm biến gia tốc điện dung

### Cảm biến gia tốc kiểu điện dung (Capacitive Accelerometers)

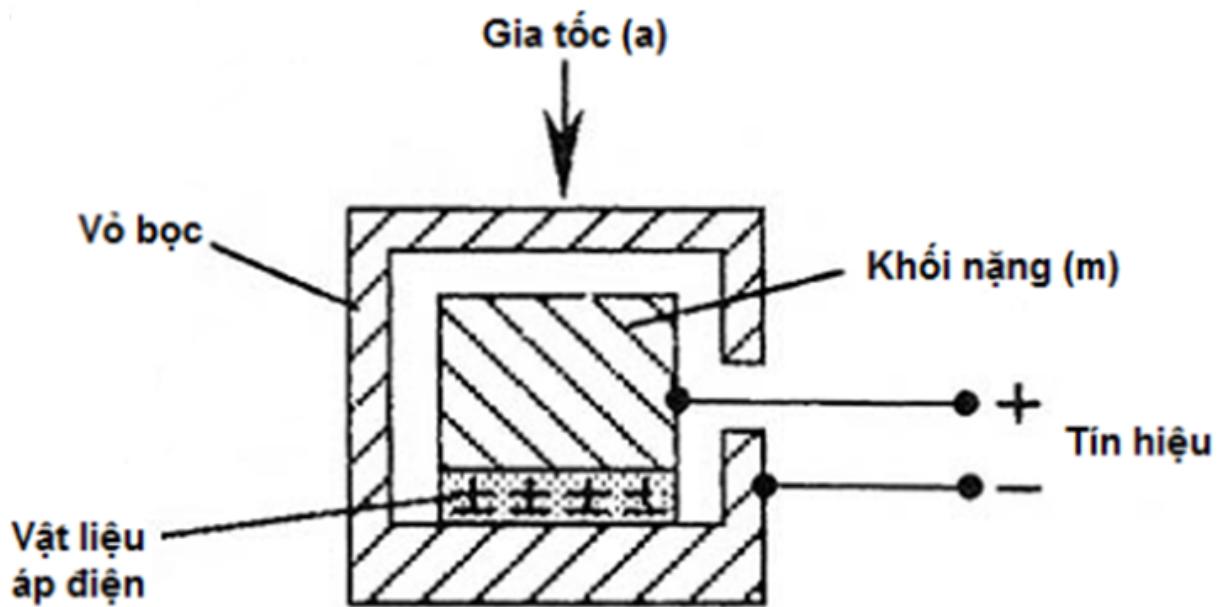


Hình 1.4: Cấu trúc cảm biến áp trớ

*Nguyên lý hoạt động:* Cảm biến gia tốc kiểu điện dung hoạt động dựa trên nguyên lý biến thiên điện dung giữa các bản cực trong cấu trúc tụ điện khi chịu tác động bởi gia tốc. Cấu hình cơ bản của cảm biến bao gồm một khối lượng vi mô (proof mass) được treo trên hệ thống lò xo vi cơ (MEMS spring system), với một đầu gắn cố định và đầu còn lại liên kết với một bản cực của tụ điện. Khi có một gia tốc tác động theo một phương xác định, khối lượng này sẽ dịch chuyển lệch khỏi vị trí cân bằng, làm thay đổi khoảng cách giữa các bản cực, từ đó gây ra sự biến thiên điện dung. Biến thiên điện dung này được phát hiện thông qua mạch đo nhạy điện dung và được chuyển đổi thành tín hiệu điện tử, tỷ lệ thuận với độ lớn của gia tốc tác động. Quá trình này cho phép cảm biến thu nhận được gia tốc theo thời gian thực với độ chính xác cao. Hình ?? minh họa nguyên lý chuyển động và thay đổi điện dung trong cấu trúc cảm biến MEMS điện dung.

### Cảm biến gia tốc kiểu áp điện trớ

Cảm biến kiểu áp điện trớ hoạt động dựa trên nguyên lý thay đổi điện trớ của các phần tử áp điện trớ khi chịu ứng suất cơ học. Trong cấu hình tiêu chuẩn, các phần tử này được gắn trên một thanh dầm (cantilever) liên kết với một khối gia trọng đặt trong môi trường cần đo. Khi có gia tốc tác động, khối gia trọng tạo ra lực quán tính khiến thanh dầm bị biến dạng, từ đó làm thay đổi điện trớ của các phần tử cảm biến. Để khuếch đại và cải thiện độ chính xác của tín hiệu, các cảm biến này thường được tích hợp trong cấu trúc mạch cầu Wheatstone. Cách bố trí này cho phép tối đa hóa tín hiệu đầu ra và tăng tỷ số tín hiệu trên nhiễu (Signal-to-Noise Ratio – SNR) của phép đo (xem Hình ??). Gia



Hình 1.5: Cấu trúc cảm biến áp điện

tốc kế áp điện trở có ưu điểm nổi bật là khả năng ghi nhận các tín hiệu thay đổi chậm, cũng như hoạt động hiệu quả trong một dải đo rộng. Nhờ đó, thiết bị có thể ghi nhận các dao động có biên độ và tần số cao, rất phù hợp cho các thử nghiệm va chạm hoặc môi trường đo động học phức tạp. Ngoài ra, cảm biến này cũng cho thấy khả năng ổn định tốt trước các thay đổi nhiệt độ của môi trường xung quanh. Tuy nhiên, hạn chế chính của tốc kế kiểu này là độ nhạy giảm khi đo tín hiệu yếu, làm giảm hiệu quả trong một số ứng dụng yêu cầu phát hiện dao động nhỏ. Bên cạnh đó, chi phí sản xuất và triển khai cao hơn đáng kể so với các loại tốc kế điện dung sử dụng công nghệ MEMS.

### Cảm biến tốc độ kiểu áp điện

Cảm biến tốc độ kiểu áp điện hoạt động dựa trên hiệu ứng áp điện của một số vật liệu đặc biệt như gốm sứ hoặc thạch anh. Khi các vật liệu này chịu ứng suất cơ học, chúng sẽ bị biến dạng và tạo ra một điện thế trên bề mặt. Lượng điện tích sinh ra tỉ lệ thuận với lực tác dụng lên cảm biến, trong khi chiều cực tính phụ thuộc vào hướng của lực. Một trong những ưu điểm nổi bật của cảm biến tốc độ áp điện so với các loại cảm biến tốc độ khác là khối lượng nhẹ và khả năng đáp ứng tần số rất cao, có thể lên đến khoảng 1 MHz. Đặc điểm này khiến chúng đặc biệt phù hợp để đo các dao động nhanh và ngắn hạn trong các môi trường khắc nghiệt hoặc ứng dụng đòi hỏi độ chính xác cao.

Tuy nhiên, cảm biến áp điện vôn có trở kháng đầu ra rất cao và chỉ tạo ra điện áp nhỏ, dễ bị suy giảm tín hiệu nếu không xử lý đúng cách. Do đó, để đảm bảo chất lượng tín hiệu và giảm thiểu hiện tượng sai số do tải (loading error), các bộ khuếch đại chuyển đổi trở kháng chuyên dụng – chẳng hạn như bộ khuếch đại điện tích – thường được tích hợp cùng hệ thống đo (xem Hình ??).

Đánh giá vị trí của con người trong lúc ngủ không phụ thuộc vào chuyển động quay xung quanh một trục, nên không nhất thiết phải tích hợp thêm con quay hồi chuyển.Thêm vào đó, những điểm nổi bật của cảm biến gia tốc trên công nghệ MEMS như việc ghép nối, tích hợp dễ dàng, trở kháng cao với mạch xử lý tín hiệu tích hợp sẵn cho phép đo biến đổi điện dung, độ nhạy cao, cho phép đáp ứng trong dải tần số đa dạng và đặc biệt là với chi phí phù hợp làm cho cảm biến gia tốc MEMS hay được lựa chọn trong phát triển thiết bị ngày nay. Trong khuôn khổ luận văn này, với tính chất của chuyển động tư thế ngủ là các chuyển động chậm với biên độ nhỏ, cảm biến MEMS kiểu điện dung được lựa chọn để tiến hành thực nghiệm.

## **1.4. Đánh giá tư thế ngủ sử dụng cảm biến gia tốc, ứng dụng công nghệ trí tuệ nhân tạo trong chẩn đoán OSA**

Vị trí là một trong bảy thông số quan trọng trong mô hình SCOPERA dùng để xác định hội chứng ngưng thở khi ngủ. Phân tích các chuyển động và tư thế trong giấc ngủ không chỉ phản ánh chất lượng giấc ngủ mà còn đóng vai trò thiết yếu trong việc đánh giá chỉ số ngưng thở – AHI. Tư thế ngủ không hợp lý có thể làm thay đổi đặc tính hô hấp và âm thanh thở, từ đó ảnh hưởng đến mức độ nghiêm trọng của OSA. Ở phần lớn bệnh nhân, mức độ nghiêm trọng của hội chứng này có xu hướng tăng rõ rệt khi họ nằm ngửa. Thông kê cho thấy, có đến 60% bệnh nhân ghi nhận chỉ số AHI tăng gấp đôi khi ngủ ở tư thế nằm ngửa – tình trạng này được gọi là hội chứng ngưng thở khi ngủ phụ thuộc tư thế (positional Obstructive Sleep Apnea – pOSA) [Unat1391].

Vì lý do đó, việc ghi nhận và phân tích tư thế ngủ đóng vai trò then chốt trong quá trình phát triển các thiết bị hỗ trợ đánh giá giấc ngủ một cách toàn diện và chính xác. Thông tin về tư thế ngủ không chỉ góp phần làm rõ mối liên hệ giữa vị trí cơ thể và mức



Hình 1.6: Các tư thế ngủ cơ bản của con người

độ nghiêm trọng của chứng ngưng thở khi ngủ, mà còn hỗ trợ cá nhân hóa phương pháp điều trị cho từng bệnh nhân. Tư thế ngủ của con người thường được phân loại thành bốn nhóm chính: nằm ngửa, nghiêng trái, nghiêng phải và nằm sấp (Hình ??) [4\_ngu]. Việc phân biệt rõ ràng các tư thế này giúp nâng cao độ chính xác trong việc phân tích ảnh hưởng của tư thế đến các chỉ số sinh lý trong giấc ngủ.

Tư thế ngủ có ảnh hưởng rõ rệt đến mức độ nghiêm trọng của hội chứng ngưng thở khi ngủ tắc nghẽn (OSA), do đó việc phân tích chi tiết từng tư thế là điều cần thiết trong các nghiên cứu và ứng dụng lâm sàng. Các tư thế ngủ khác nhau có thể tác động trực tiếp đến cơ chế tắc nghẽn đường thở, làm thay đổi tần suất và mức độ nghiêm trọng của các đợt ngưng thở. Trong số đó, các tư thế nằm nghiêng (bao gồm nghiêng trái và nghiêng phải) được ghi nhận là có khả năng cải thiện triệu chứng OSA ở nhiều bệnh nhân nhờ hạn chế tình trạng sụp đổ của đường thở trên so với tư thế nằm ngửa. Việc duy trì tư thế nằm nghiêng giúp giảm hiện tượng ngáy và hỗ trợ lưu thông khí thở tốt hơn. Tuy nhiên, sự khác biệt rõ ràng về hiệu quả giữa tư thế nghiêng trái và nghiêng phải vẫn chưa được xác định chắc chắn, và cần được đánh giá dựa trên đặc điểm cá nhân và bệnh lý kèm theo

của từng người bệnh. Tư thế nằm sấp có thể làm giảm nguy cơ tắc nghẽn đường thở do trọng lực kéo các mô mềm, bao gồm cả lưỡi, về phía trước. Tuy nhiên, tư thế này không phù hợp với mọi đối tượng, đặc biệt là những người có thói quen che mặt hoặc miếng băng gối, vì có thể làm hạn chế luồng khí hít vào. Ngoài ra, ở những bệnh nhân có vấn đề về cột sống cổ, tư thế nằm sấp có thể gây ra căng thẳng không cần thiết và làm trầm trọng thêm tình trạng cơ xương khớp. Ngược lại, tư thế nằm ngửa thường liên quan đến nguy cơ cao hơn đối với các đợt ngưng thở khi ngủ. Trong tư thế này, các mô mềm của đường thở trên – đặc biệt là lưỡi – dễ bị trọng lực kéo lùi về phía sau, gây hẹp hoặc tắc nghẽn đường thở. Do đó, nằm ngửa được xem là tư thế dễ làm trầm trọng thêm OSA và nên được cân nhắc hạn chế trong chiến lược quản lý bệnh [LANDRY2023101847].

Sự phát triển nhanh chóng của trí tuệ nhân tạo (AI) trong lĩnh vực y học đã mở ra nhiều hướng tiếp cận mới trong việc đánh giá tư thế ngủ và chẩn đoán OSA. Các hệ thống AI đang dần chứng minh hiệu quả vượt trội trong việc xử lý dữ liệu cảm biến lớn và phức tạp, từ đó cung cấp các phân tích chính xác về hành vi giấc ngủ của bệnh nhân. Các mô hình học máy được huấn luyện trên tập dữ liệu cảm biến từ accelerometer, gyroscope hoặc thiết bị đeo thông minh có thể tự động phân loại các tư thế ngủ theo thời gian thực, với độ chính xác lên đến trên 90% trong nhiều nghiên cứu gần đây [Sleep\_Posture\_Detection][Vu2025SleepPosition][HOANG2025116309].

Nhờ vào khả năng học và tự hiệu chỉnh, các thuật toán này có thể phân biệt hiệu quả giữa các trạng thái nằm nghiêng trái, nghiêng phải, nằm ngửa và nằm sấp – ngay cả khi có sự biến đổi nhẹ về góc độ hoặc cử động cơ thể. Hơn nữa, AI còn cho phép tích hợp thông tin về tư thế ngủ với các chỉ số sinh lý khác như nhịp tim, nhịp thở, SpO và dữ liệu âm thanh để xây dựng mô hình chẩn đoán OSA đa chiều. Việc kết hợp các nguồn dữ liệu này giúp phát hiện chính xác các giai đoạn ngưng thở và giảm thở, đồng thời đánh giá được mức độ ảnh hưởng của từng tư thế đến tình trạng hẹp đường thở trong khi ngủ. Đây là một bước tiến quan trọng hướng đến cá nhân hóa chẩn đoán và điều trị OSA – điều mà các phương pháp truyền thống như đa ký giấc ngủ (polysomnography) còn nhiều hạn chế do chi phí cao và điều kiện thực hiện phức tạp. Đặc biệt, các hệ thống AI có thể được triển khai trong các thiết bị đeo thông minh tại nhà, hỗ trợ theo dõi lâu dài và liên tục mà không gây xâm lấn hay gián đoạn giấc ngủ. Nhờ đó, dữ liệu thu thập được phản ánh chính xác hơn về hành vi giấc ngủ trong môi trường tự nhiên của người bệnh, từ đó nâng cao giá trị lâm sàng của các kết quả phân tích. Ngoài ra, sự tích hợp AI trong các thiết bị

di động, cùng với công nghệ điện toán biên (edge computing), có thể cho phép xử lý dữ liệu tại chỗ và phản hồi thời gian thực – mở ra tiềm năng to lớn trong việc sàng lọc, theo dõi và cá nhân hóa chiến lược quản lý OSA.

## **Chương 2**

# **LỰA CHỌN VÀ XÂY DỰNG THIẾT BỊ**

### **2.1. Nghiên cứu, lựa chọn cảm biến**

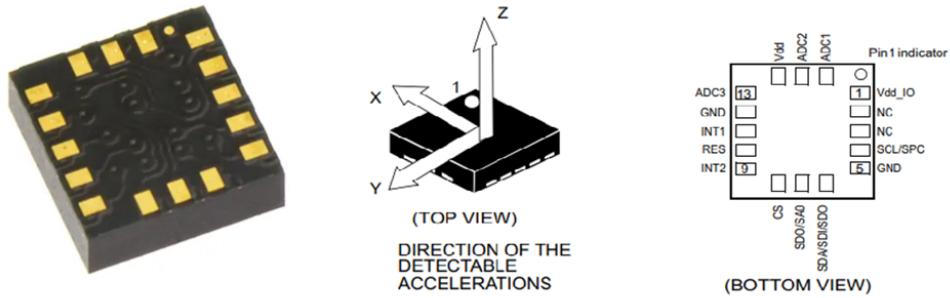
Trong tự nhiên, các chuyển động vật lý được phân loại thành ba nhóm chính: chuyển động đột ngột, chuyển động chậm và chuyển động dao động điều hòa. Đối với chuyển động của con người trong giấc ngủ, đây chủ yếu là chuyển động chậm, diễn ra ở cả hai giai đoạn: (1) giấc ngủ không chuyển động mắt nhanh (Non-Rapid Eye Movement – NREM) và (2) giấc ngủ có chuyển động mắt nhanh (Rapid Eye Movement – REM). Như đã trình bày trong Chương I, hiện nay có nhiều loại cảm biến gia tốc MEMS được ứng dụng rộng rãi trong lĩnh vực theo dõi tư thế ngủ. Sau quá trình khảo sát và đánh giá các tiêu chí kỹ thuật, tác giả đã tiến hành thực nghiệm với dòng cảm biến gia tốc phổ biến là LIS3DH [**LIS3DH**] và 1 dòng LSM6DS3 [**st\_lsm6ds3\_2017**] trong giai đoạn triển khai trên chip do công ty STMicroelectronics sản xuất. Mỗi loại cảm biến có những ưu điểm riêng, phù hợp với từng mục tiêu thiết kế cụ thể

Tiêu chí đầu tiên được xem xét là độ nhạy, thể hiện tỉ lệ giữa tín hiệu điện đầu ra và gia tốc g. Độ nhạy thường được biểu diễn bằng các đơn vị như mV/g, count/g hoặc LSB/g (Least Significant Bit per g), và có mối quan hệ tỉ lệ nghịch với biên độ tối đa mà cảm biến có thể đo được. Độ nhạy cao cho phép phát hiện các chuyển động nhỏ, vốn rất đặc trưng trong giấc ngủ. Bên cạnh đó, loại tín hiệu thay đổi cũng đóng vai trò quan

trọng trong việc xác định cảm biến phù hợp. Trong ba loại nguyên lý phổ biến dùng để chế tạo cảm biến gia tốc là điện dung, điện trở và điện áp, thì cảm biến dựa trên nguyên lý điện dung và áp điện được đánh giá là phù hợp hơn cả cho các ứng dụng theo dõi chuyển động chậm. Cảm biến áp điện đặc biệt hiệu quả trong việc ghi nhận các dao động điều hòa, trong khi cảm biến áp điện trở lại thích hợp hơn để phát hiện các chuyển động đột ngột. Tiêu chí tiếp theo là biên độ đo lường, tức khoảng gia tốc mà cảm biến có thể ghi nhận chính xác. Nếu gia tốc vượt ngoài khoảng này, tín hiệu sẽ bị sai lệch. LIS3DH cho phép lập trình dải đo từ  $\pm 2g$  đến  $\pm 16g$ . Cần lưu ý rằng biên độ càng lớn thì độ nhạy và độ phân giải càng giảm. Ví dụ, với cảm biến 16-bit có tổng cộng 65.536 mức giá trị, nếu chọn dải đo  $\pm 100g$ , thì độ phân giải sẽ tương đương khoảng 0,003g. Một yếu tố quan trọng khác là số trục đo. LIS3DH hỗ trợ đo trên ba trục X, Y, Z, đáp ứng tiêu chuẩn ba bậc tự do (3 Degrees of Freedom – DOF), đủ để xác định tư thế và vị trí trong không gian ba chiều. Số bậc tự do càng cao thì lượng thông tin chuyển động thu thập được càng phong phú và chính xác hơn, tuy nhiên cũng kéo theo sự phức tạp trong thiết kế và xử lý dữ liệu. Mức tiêu thụ năng lượng là một tiêu chí không thể bỏ qua, đặc biệt trong các hệ thống đeo cá nhân. Cảm biến có công suất tiêu thụ thấp sẽ kéo dài thời gian hoạt động của thiết bị, đồng thời cho phép sử dụng pin nhỏ hơn, từ đó giảm kích thước tổng thể và tăng tính di động.

Cuối cùng, yếu tố giá thành cũng được cân nhắc kỹ lưỡng. Các bộ phát triển thương mại thường có giá cao do tích hợp nhiều chức năng phần cứng và phần mềm. Tuy nhiên, đối với ứng dụng chuyên biệt như theo dõi tư thế ngủ, việc tự phát triển phần cứng và phần mềm tùy biến có thể giúp tối ưu hóa chi phí đáng kể. Trong khuôn khổ luận văn này, tác giả sử dụng bộ phát triển có sẵn để tiến hành đánh giá kỹ thuật đo lường và thu thập dữ liệu. Giai đoạn phát triển phần cứng tùy biến sẽ được thực hiện trong các nghiên cứu tiếp theo.

LIS3DH với thang đo đầy đủ khá rộng cho phép người phát triển có thể lựa chọn các giá trị  $\pm 2g$  /  $\pm 4g$  /  $\pm 8g$  /  $\pm 16g$  và có khả năng đo gia tốc với tần số ở lối ra từ 1 Hz đến 5.3 kHz. Thiết bị có thể được cấu hình để tạo ra tín hiệu ngắn bằng cách sử dụng hai sự kiện đánh thức / rơi tự do theo quán tính độc lập cũng như theo vị trí của chính thiết bị. Người dùng cuối có thể lập trình ngưỡng và thời gian của bộ tạo ngắn. LIS3DH tích hợp bộ đệm 32 cấp vào ra tuần tự (FIFO) cho phép người dùng lưu trữ dữ liệu nhằm hạn chế sự can thiệp của bộ xử lý chủ. LIS3DH được đảm bảo hoạt động trong phạm vi nhiệt



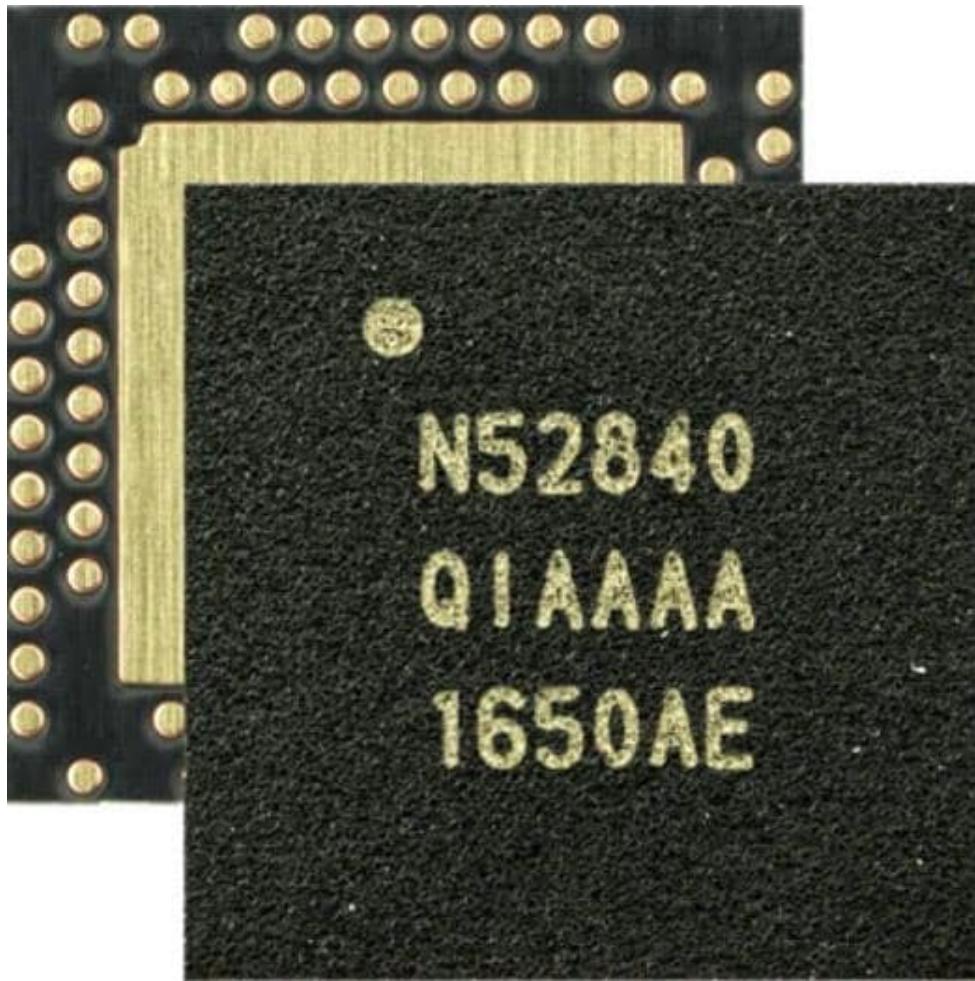
Hình 2.1: Cảm biến gia tốc LIS3DH và sơ đồ chân kết nối

độ mở rộng từ -40°C đến +85°C. LIS3DH có kích thước nhỏ gọn, thích hợp trong thiết kế mạch đo nhỏ gọn, Hình ???. Thiết bị hỗ trợ dải điện áp cung cấp rộng, dao động từ 1.71 V đến 3.6 V, cho phép linh hoạt trong thiết kế nguồn. Ngoài ra, cảm biến còn tích hợp nguồn cung cấp I/O độc lập ở mức 1.8 V, đồng thời vẫn đảm bảo khả năng tương thích với các hệ thống điện áp khác nhau. Một trong những ưu điểm nổi bật của LIS3DH là chế độ tiêu thụ dòng điện cực thấp, chỉ khoảng 2 A trong chế độ tiết kiệm năng lượng, điều này rất quan trọng đối với các hệ thống hoạt động liên tục trong thời gian dài. Thiết bị sử dụng giao diện kỹ thuật số I2C/SPI cho đầu ra dữ liệu, giúp đơn giản hóa việc truyền nhận thông tin giữa cảm biến và bộ xử lý trung tâm. Cảm biến cung cấp đầu ra dữ liệu với độ phân giải 10 bit, cho phép ghi nhận chính xác các thay đổi nhỏ trong chuyển động. Bên cạnh đó, LIS3DH còn được trang bị hai bộ ngắt lập trình độc lập, hỗ trợ phát hiện các chuyển động đặc biệt như rơi tự do và chuyển động nhanh, đồng thời tích hợp tính năng phát hiện định hướng không gian 4D/6D, rất hữu ích trong việc phân loại tư thế ngủ. Thiết bị này đáp ứng các tiêu chuẩn môi trường nghiêm ngặt như ECOPACK®, RoHS và "Green", bảo đảm thân thiện với môi trường và an toàn cho người sử dụng. Trong ngữ cảnh theo dõi tư thế ngủ, khi người đang ngủ thay đổi vị trí, gia tốc cơ thể thường dao động trong phạm vi  $\pm 2g$ . Với độ phân giải đầu ra 10 bit, tức 1024 mức chia, hệ thống có thể đạt độ phân giải lý thuyết khoảng 0.004g. Độ phân giải này là hoàn toàn phù hợp để ghi nhận các thay đổi chậm và nhỏ đặc trưng trong quá trình chuyển đổi tư thế khi ngủ, từ đó hỗ trợ hiệu quả cho các mô hình phân tích bằng trí tuệ nhân tạo.

## 2.2. Nghiên cứu, lựa chọn vi xử lý

Với sự phát triển vượt bậc và đa dạng của công nghệ chế tạo, có rất nhiều cấu hình phần cứng được nhiều nhóm tác giả lựa chọn phù hợp với các mục đích khác nhau. Trong đó, [p\_1] các tác giả đã sử dụng máy tính đơn Raspberry Pi kết hợp các điện trở cảm biến lực để phát hiện 4 tư thế ngủ với sự lấy nhãn từ video theo dõi người bệnh trong suốt quá trình lấy mẫu. Kwasnicki và cộng sự đã phát triển hệ thống ngủ có thể đeo (wearable sleep system) sử dụng bộ xử lý công suất thấp TI MSP430 và mô-đun RF Chipcon CC2420 cho truyền thông không dây kết hợp với cảm biến gia tốc 3 trục ADXL330, con quay hồi chuyển InvenSense ITG-3200, Honeywell HMC5843 để đo từ trường xác định 99.5% chính xác 4 tư thế ngủ [p\_2]. Tuy nhiên, các thiết bị vẫn yêu cầu một nguồn năng lượng khiếu nại cho tính liên tục bị hạn chế đáng kể. I.Yun và cộng sự đã phát triển thiết bị theo dõi tư thế ngủ của trẻ nhỏ sử dụng vi xử lý ATmega328P-PU cùng module Bluetooth kết hợp cảm biến gia tốc ADXL335 được đặt trên bụng đã nhưng lựa chọn về mặt cấu hình thiết bị và chế tạo ra mạch cung cấp năng lượng cho những thành phần cần thiết [p\_3]. Từ đó, giảm thiểu đáng kể mức tiêu thụ năng lượng và vẫn giữ nguyên độ chính xác nhưng khá bất tiện cho trẻ nhỏ.

Sau quá trình khảo sát, đánh giá và so sánh giữa các lựa chọn vi xử lý phổ biến hiện nay, tác giả quyết định lựa chọn vi xử lý nRF52840 [nrf] cho hệ thống được đề xuất, dựa trên những ưu điểm nổi bật như kích thước nhỏ gọn, mức tiêu thụ năng lượng thấp và khả năng tích hợp Bluetooth năng lượng thấp (Bluetooth Low Energy – BLE). nRF52840 là vi xử lý cao cấp nhất trong dòng họ nRF52 do hãng Nordic Semiconductor phát triển. Đây là một hệ thống tích hợp trên một vi mạch (System-on-Chip – SoC), được thiết kế tối ưu cho các ứng dụng yêu cầu mức tiêu thụ năng lượng cực thấp, đặc biệt là trong các mạng không dây tầm ngắn. Vi xử lý này tích hợp bộ thu phát đa giao thức hoạt động ở băng tần 2.4 GHz, sử dụng kiến trúc CPU Arm Cortex-M4F với tốc độ xung nhịp 64 MHz, cùng bộ xử lý dấu phẩy động (FPU) và dung lượng bộ nhớ gồm 1 MB bộ nhớ Flash và 256 KB RAM. Bên cạnh đó, nRF52840 hỗ trợ chuẩn Bluetooth 5.3 với khả năng giao tiếp đa giao thức, cho phép tăng cường phạm vi truyền dữ liệu, cải thiện tốc độ truyền và giảm tiêu thụ năng lượng. Bộ tính năng bảo mật được tích hợp một cách đầy đủ, đáp ứng các yêu cầu ngày càng cao trong bảo vệ dữ liệu và truyền thông không dây. Vi xử lý này còn sở hữu khả năng quản lý năng lượng linh hoạt, có thể hoạt động trong dải điện áp



Hình 2.2: Nordic Semiconductor NRF52840

rộng từ +5.5V xuống đến +1.7V, cho phép thiết bị hoạt động trực tiếp từ nguồn pin hoặc cổng USB. Về khả năng giao tiếp ngoại vi, nRF52840 cung cấp cấu hình đa dạng, bao gồm tối đa hai giao tiếp I2C, bốn giao tiếp SPI ở chế độ master và ba SPI ở chế độ slave, cùng bốn kênh PWM tích hợp EasyDMA. Ngoài ra, thiết bị còn hỗ trợ năm bộ định thời 32-bit, phù hợp với các ứng dụng đòi hỏi độ chính xác cao trong xử lý thời gian thực.

### 2.2.1. Vi điều khiển ARM Cortex-M4

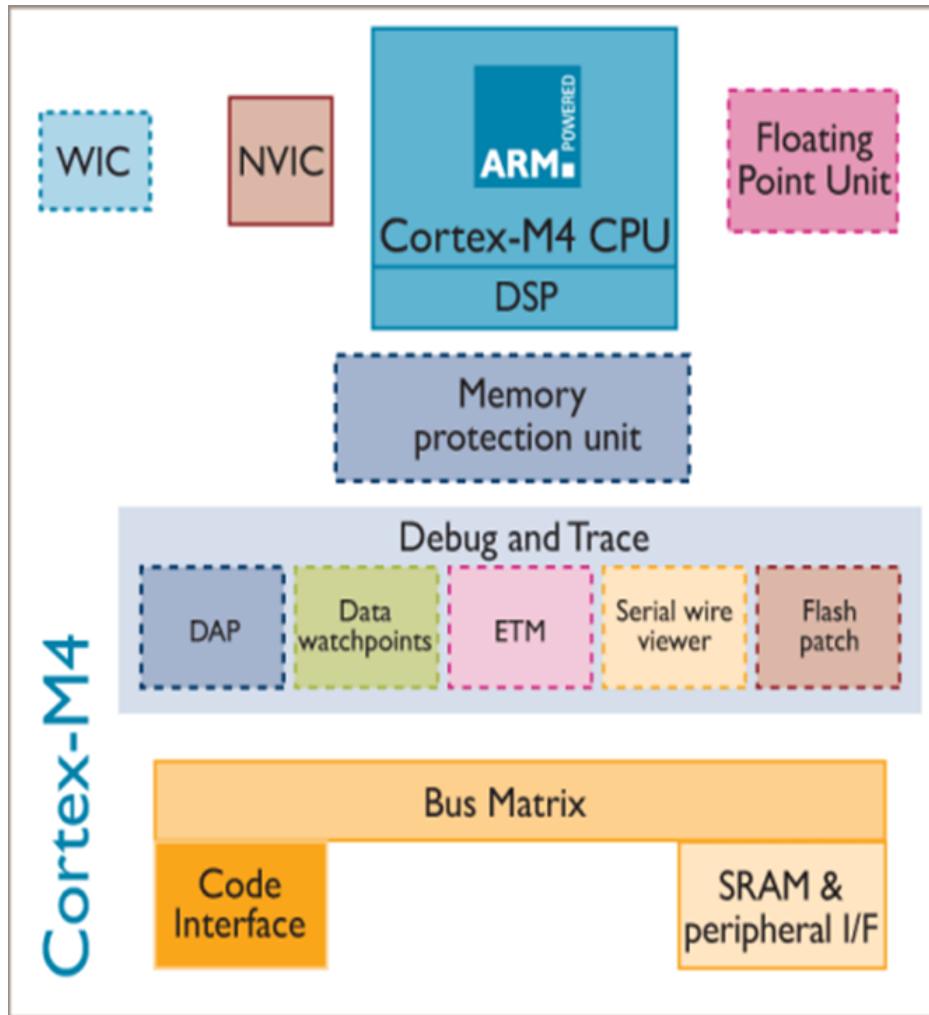
Kiến trúc ARM có nhiều dòng vi xử lý khác nhau, được phát triển và nâng cấp liên tục nhằm đáp ứng nhu cầu đa dạng trong lĩnh vực công nghệ nhúng. Trong đó, dòng Cortex-M thuộc kiến trúc ARMv7 đã trở thành nền tảng phổ biến cho các hệ thống nhúng sử dụng vi điều khiển nhờ vào hiệu suất cao, khả năng mở rộng và mức tiêu thụ năng lượng tối ưu. Dòng Cortex-M bao gồm nhiều phiên bản như Cortex-M0, Cortex-M0+, Cortex-

M1, Cortex-M3, Cortex-M4 và Cortex-M7, mỗi phiên bản được thiết kế để phục vụ cho các mức độ yêu cầu hiệu năng khác nhau [[arm\\_cortex\\_m\\_comparison](#)]. Các vi xử lý thuộc họ Cortex-M chủ yếu được ứng dụng trong các hệ thống nhúng thời gian thực, nơi yêu cầu sự cân bằng giữa hiệu suất xử lý, tiêu thụ năng lượng và chi phí. Một số vi xử lý ARM khác, không thuộc họ Cortex-M, được sử dụng trong các thiết bị hiệu suất cao như điện thoại thông minh và máy tính bảng, vốn yêu cầu cấu hình phần cứng mạnh hơn và khả năng xử lý đa tác vụ cao hơn. Ngoài ra, hiện nay ứng dụng trí tuệ nhân tạo (AI) tại thiết bị biên (Edge AI) đang ngày càng phổ biến, đặc biệt trong các lĩnh vực như nhà thông minh, thiết bị đeo, giám sát an ninh và công nghiệp 4.0. Với khả năng xử lý tín hiệu số (DSP) và hỗ trợ các mạng nơ-ron nhỏ gọn, các vi xử lý Cortex-M, đặc biệt là dòng Cortex-M4, đang được khai thác để triển khai các mô hình học sâu nhẹ (tinyML) ngay trên vi điều khiển [[electronics11162545](#)][[applicationCortexM4](#)].

Nhờ vào đặc điểm kỹ thuật như vậy, Cortex-M4 được ứng dụng rộng rãi trong nhiều lĩnh vực khác nhau, bao gồm điều khiển động cơ, công nghiệp ô tô, hệ thống quản lý năng lượng, âm thanh nhúng và các hệ thống tự động hóa công nghiệp. Tính linh hoạt và hiệu quả về hiệu suất – năng lượng giúp Cortex-M4 trở thành một trong những vi xử lý được ưa chuộng nhất trong phát triển hệ thống nhúng hiện đại.

Theo tài liệu [[cortexM4](#)], bộ vi xử lý Cortex-M4 là một vi xử lý 32-bit sử dụng kiến trúc tập lệnh giảm (RISC), được thiết kế theo kiến trúc Harvard, trong đó bus dữ liệu và bus lệnh được tách biệt nhằm tối ưu tốc độ truyền tải và xử lý. Vi xử lý này hỗ trợ đầy đủ tập lệnh Thumb-1 (16-bit) cũng như Thumb-2 (kết hợp 16-bit và 32-bit), mang lại sự linh hoạt trong việc mã hóa lệnh và hiệu quả bộ nhớ. Về mặt hiệu năng, Cortex-M4 đạt được từ 1,25 đến 1,95 DMIPS trên mỗi MHz (Dhrystone Million Instructions Per Second per MHz), cho thấy khả năng xử lý tốt trong các ứng dụng nhúng yêu cầu độ chính xác cao. Bộ xử lý hỗ trợ lên đến 240 tín hiệu ngắn, bao gồm các ngắn ngoại lệ không chẩn được (Non-Maskable Interrupts – NMI), cùng với khả năng cấu hình từ 8 đến 256 mức ngắn ưu tiên khác nhau. Điều này giúp hệ thống xử lý hiệu quả các tác vụ thời gian thực, đặc biệt trong môi trường nhúng có nhiều ngắn cạnh tranh xảy ra đồng thời.

Kết nối bus được mô tả trong Hình ?? cho phép truyền dữ liệu đồng thời trên nhiều bus khác nhau, đồng thời cung cấp khả năng quản lý truyền dữ liệu hiệu quả, chẳng hạn như sử dụng bộ đệm ghi và điều khiển hướng bit hoạt động (bit-banding). Hệ thống cũng



Hình 2.3: Thành phần chính của vi điều khiển Cortex-M4

có thể bao gồm các cầu bus (bus bridges) nhằm kết nối nhiều loại bus vào một mạng duy nhất sử dụng chung không gian bộ nhớ. Ngoài ra, bộ xử lý được trang bị hệ thống hỗ trợ gỡ lỗi tích hợp, bao gồm khả năng kiểm soát gỡ lỗi, thiết lập điểm ngắt (breakpoint) chương trình và điểm theo dõi dữ liệu (watchpoint). Khi xảy ra sự kiện gỡ lỗi, hệ thống có thể tạm dừng trạng thái hoạt động của lỗi xử lý để phục vụ việc phân tích và xử lý lỗi.

Bên cạnh đó, kiến trúc Cortex-M4 tích hợp Bộ điều khiển ngắt vectored lồng nhau (Nested Vectored Interrupt Controller – NVIC) với khả năng hỗ trợ lên đến 240 tín hiệu yêu cầu ngắt, bao gồm cả ngắt không chắn được (NMI). NVIC hỗ trợ xử lý ngắt lồng nhau một cách tự động bằng cách so sánh mức ưu tiên giữa các yêu cầu ngắt với mức ưu tiên hiện tại đang được xử lý.

Đối với các ứng dụng yêu cầu tiết kiệm năng lượng, hệ thống còn được trang bị bộ

đánh thức ngắt (Wake-up Interrupt Controller – WIC), cho phép đưa bộ vi điều khiển vào chế độ nghỉ bằng cách tắt hầu hết các thành phần không cần thiết, đồng thời duy trì khả năng đánh thức hệ thống khi phát hiện một yêu cầu ngắt. Ngoài ra, cơ chế bảo vệ bộ nhớ cũng được tích hợp nhằm đảm bảo an toàn cho hệ thống, ví dụ như chỉ cho phép truy cập đọc tại một số vùng bộ nhớ hoặc ngăn người dùng truy cập vào các vùng dữ liệu đặc quyền của hệ điều hành hoặc ứng dụng hệ thống.

### 2.2.2. Bluetooth năng lượng thấp

	nRF52805	nRF52810	nRF52811	nRF52820	nRF52832	nRF52833	nRF52840	nRF5340
Bluetooth 5.3	X	X	X	X	X	X	X	X
Bluetooth 2 Mbps	X	X	X	X	X	X	X	X
Bluetooth Long Range			X	X		X	X	X
Bluetooth Direction Finding				X	X		X	
Bluetooth LE Audio								X
Bluetooth mesh				X	X	X	X	X
Thread			X	X		X	X	X
Zigbee				X		X	X	X
Matter							X	X

Hình 2.4: Các kiểu kết nối không dây trong họ chip nRF52

Bluetooth năng lượng thấp (Bluetooth Low Energy – BLE) là một chuẩn kết nối không dây chi phí thấp, được thiết kế tối ưu cho các ứng dụng tiêu thụ điện năng thấp, đặc biệt phù hợp với các thiết bị hoạt động bằng pin kích thước nhỏ. BLE hoạt động trong băng tần ISM 2.4 GHz, hỗ trợ thông lượng ứng dụng lên đến 1.4 Mbps, cho phép

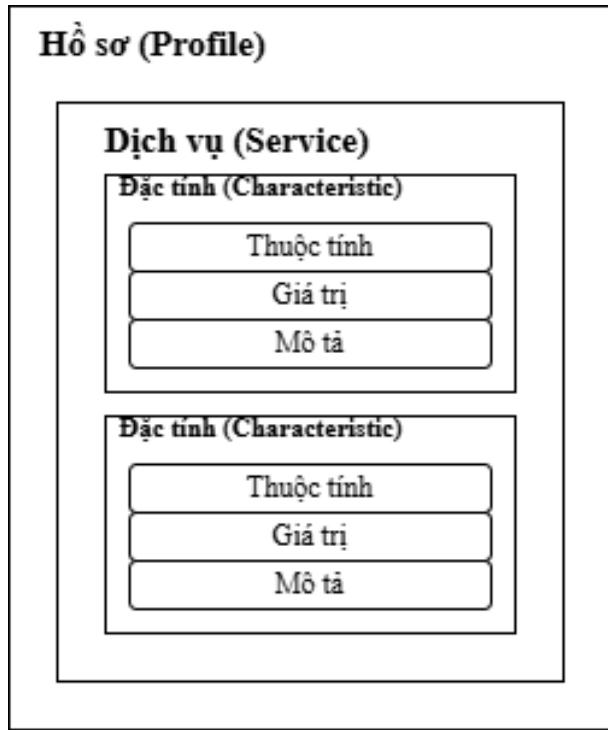
truyền dữ liệu hiệu quả trong khi vẫn duy trì mức tiêu thụ năng lượng tối thiểu.

Về mặt bảo mật, BLE tích hợp các cơ chế mã hóa và xác thực nhằm đảm bảo tính bí mật, toàn vẹn và riêng tư của dữ liệu truyền qua mạng. Công nghệ này đã trở thành một phần tiêu chuẩn trong hầu hết các thiết bị di động hiện đại như smartphone, máy tính bảng, và laptop, đồng thời được hỗ trợ đầy đủ trên các hệ điều hành phổ biến bao gồm iOS, Android, macOS, Windows 10 và Linux. Bluetooth 5 là bước phát triển đột phá tiếp theo kể từ khi BLE được giới thiệu trong chuẩn Bluetooth 4.0, mang đến hàng loạt cải tiến đáng kể giúp mở rộng phạm vi ứng dụng và nâng cao hiệu suất hệ thống. Một trong những cải tiến nổi bật là chế độ 2 Mbps, cho phép tăng gấp đôi tốc độ truyền lý thuyết, tương ứng với thông lượng thực tế lên đến 1.4 Mbps. Quan trọng hơn, chế độ này còn giúp giảm đáng kể mức tiêu thụ năng lượng – cụ thể là giảm một nửa năng lượng tiêu thụ trên mỗi bit dữ liệu – từ đó kéo dài thời gian hoạt động của thiết bị hoặc cho phép sử dụng các nguồn năng lượng nhỏ và chi phí thấp hơn [BLE].

Bên cạnh đó, tính năng Advertising Extensions (mở rộng quảng cáo) đã cách mạng hóa cơ chế phát sóng của BLE. Các gói quảng cáo giờ đây có thể chứa lượng dữ liệu gấp 8 lần so với phiên bản trước, cho phép truyền tải các khối dữ liệu lớn hơn mà không cần thiết lập kết nối ngay lập tức. Đồng thời, các gói quảng cáo có thể được xâu chuỗi để tạo thành các tập tin quảng cáo phức hợp. Tính năng lựa chọn kênh được tối ưu hóa giúp tăng cường độ ổn định và khả năng chống nhiễu trong các môi trường có mật độ thiết bị cao. Đặc biệt, chế độ Long Range mở rộng đáng kể phạm vi truyền thông của BLE, cho phép các thiết bị duy trì kết nối trong toàn bộ không gian của một ngôi nhà thông minh hoặc trong các ứng dụng IoT công nghiệp quy mô vừa và nhỏ.

Ngoài các tính năng nổi bật đã đề cập, vi xử lý nRF52840 còn hỗ trợ Bluetooth Mesh, một công nghệ mạng lưới tiên tiến được phát triển nhằm mở rộng đáng kể khả năng truyền thông của Bluetooth Low Energy (BLE). Bluetooth Mesh cho phép thiết lập mạng truyền thông nhiều-đến-nhiều (many-to-many), hoạt động ổn định trong các hệ thống có thể lên đến hàng nghìn nút mạng, phù hợp với các ứng dụng IoT quy mô lớn như nhà thông minh, chiếu sáng công nghiệp, giám sát môi trường và hệ thống cảm biến phân tán.

Bluetooth Mesh sử dụng BLE làm lớp truyền tải vật lý, trong đó các thông điệp được đóng gói trong các gói quảng cáo (advertising packets) hoặc gói GATT – còn được gọi là gói cấu hình thuộc tính chung (Generic Attribute Profile – GATT). Khi một thiết



Hình 2.5: Cấu trúc của GATT

bị di động hoặc máy chủ trung tâm kết nối với một nút BLE bất kỳ trong mạng Mesh, nút này có thể chuyển tiếp (relay) thông điệp đến các nút khác trong mạng theo một kiến trúc định tuyến phân tán, như minh họa trong Hình ??.

Cấu trúc logic của BLE được tổ chức dựa trên mô hình GATT. GATT định nghĩa cách hai thiết bị BLE truyền dữ liệu qua lại bằng các thành phần logic gọi là dịch vụ (services) và đặc tính (characteristics). Giao tiếp GATT dựa trên Attribute Protocol (ATT) – một giao thức định hướng dữ liệu, nơi các dịch vụ và đặc tính được lưu trữ và nhận dạng bằng các mã định danh (UUID) có độ dài 16-bit hoặc 128-bit. Hình ?? minh họa cấu trúc phân cấp của GATT, trong đó một thiết bị ngoại vi BLE cung cấp nhiều dịch vụ, mỗi dịch vụ lại bao gồm một hoặc nhiều đặc tính có thể được đọc, ghi hoặc thông báo (notify) tùy theo cấu hình.

Điều đặc biệt quan trọng trong giao tiếp GATT là tính chất kết nối độc quyền (exclusive connection): tại một thời điểm nhất định, một thiết bị ngoại vi chỉ có thể kết nối với một thiết bị trung tâm duy nhất. Ngay khi kết nối được thiết lập, thiết bị ngoại vi sẽ ngừng quảng cáo, khiến các thiết bị khác không thể phát hiện hoặc kết nối với nó cho đến khi kết nối hiện tại bị ngắt.

Hồ sơ (Profile) là tập hợp các dịch vụ được xác định trước, được tiêu chuẩn hóa bởi tổ chức Bluetooth SIG hoặc do nhà phát triển thiết kế tùy chỉnh. Mỗi dịch vụ đại diện cho một thực thể logic trong hệ thống BLE, chứa một hoặc nhiều đặc tính, là đơn vị dữ liệu cơ bản nhất. Một đặc tính bao gồm giá trị chính (value), mô tả (description), và các thuộc tính điều khiển quyền truy cập như chỉ đọc, chỉ ghi hoặc hỗ trợ thông báo.

Chẳng hạn, một ứng dụng BLE có thể triển khai Dịch vụ UART tùy chỉnh, bao gồm hai đặc tính đại diện cho kênh truyền (TX) và kênh nhận (RX), trong đó một đặc tính chỉ cho phép đọc và đặc tính còn lại cho phép ghi từ phía thiết bị trung tâm.

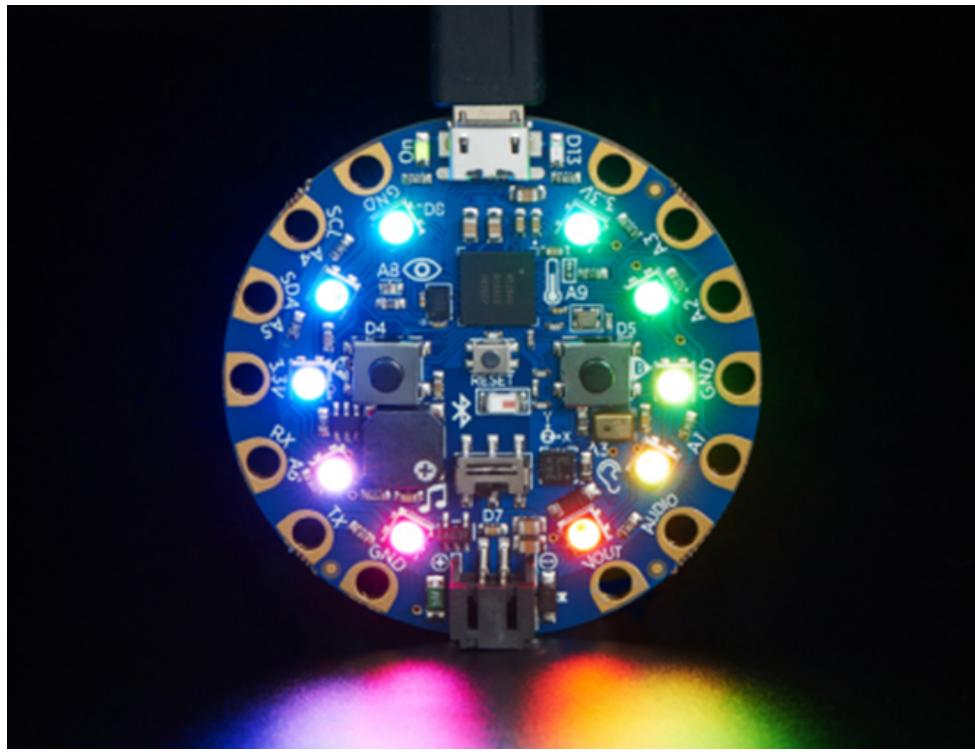
Với các khả năng kỹ thuật tiên tiến như trên – bao gồm hỗ trợ Bluetooth Mesh, giao tiếp GATT hiệu quả, khả năng quản lý năng lượng linh hoạt, bảo mật tích hợp và khả năng xử lý tín hiệu số – nRF52840 được xem là lựa chọn phù hợp và chiến lược trong khuôn khổ triển khai hệ thống được đề xuất trong luận văn này.

### 2.2.3. Thiết bị thử nghiệm

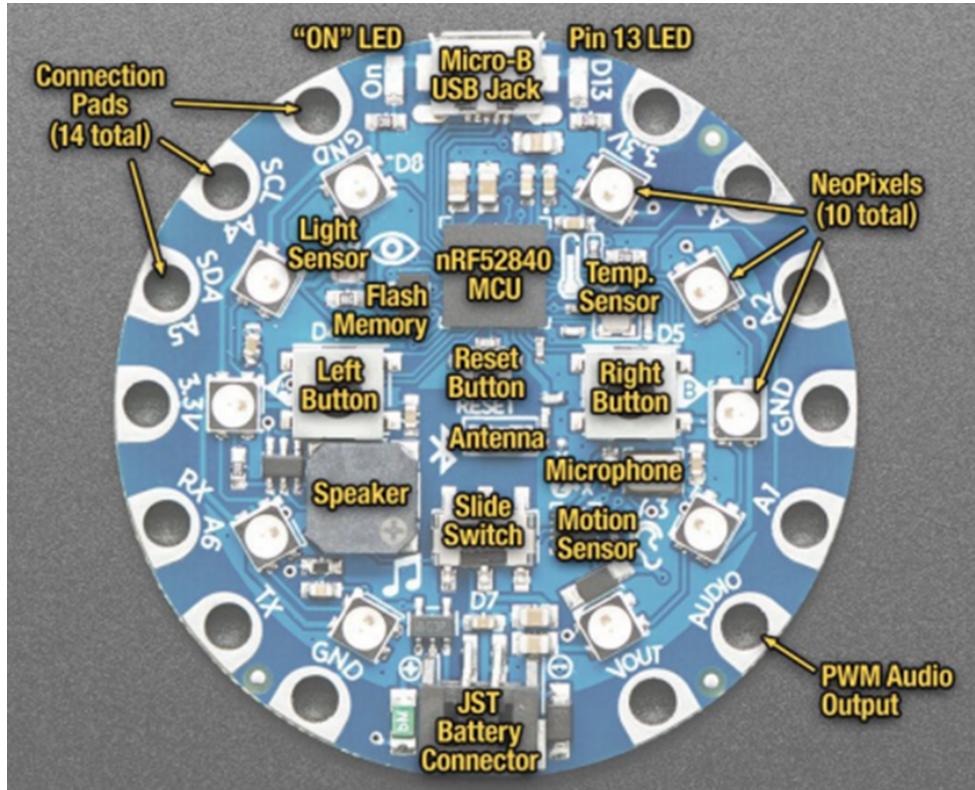
Trong khuôn khổ của khoá luận, vì lý do thời gian và sự an toàn nên tác giả sử dụng bộ kit Adafruit PlayGround để thực nghiệm. Trong bộ kit có 1 cảm biến gia tốc MEMS LIS3DH ở chính giữa. Cảm biến gia tốc này thường dùng trong điện thoại và các thiết bị điện tử khác có thể cảm nhận độ nghiêng, trọng lực, chuyển động và các hiệu ứng 'chạm'. Thông tin mà tác giả có được, chi phí cho mỗi một kit từ Adafruit khoảng 25 USD [[ada\\_price](#)]. LIS3DH được kết nối với các chân SPI phần cứng và có chân chọn chip (CS) trên chân số 8 và đầu ra ngắn tùy chọn trên chân số 7 (còn được gọi là IRQ 4). Có thể phát hiện các giá trị gia tốc theo 3 trục X, Y và Z. Giá trị dương là mặt màu của kit. Trục X là hướng tới giắc cắm USB. Trục Y trực hướng sang bên trái. Đối với Z là hướng thẳng hướng lên.

#### Các thành phần trên bộ kit Adafruit Circuit PlayGround[[ada\\_overview](#)]:

- 1 x bộ xử lý nRF52840 Cortex M4 với hỗ trợ Bluetooth Low Energy
- 10 x NeoPixels mini, mỗi cái có thể hiển thị bất kỳ màu nào
- 1 x Cảm biến chuyển động (gia tốc kế ba trục LIS3DH với phát hiện chạm, phát hiện rơi tự do)



Hình 2.6: Hình ảnh kit Adafruit PlayGround

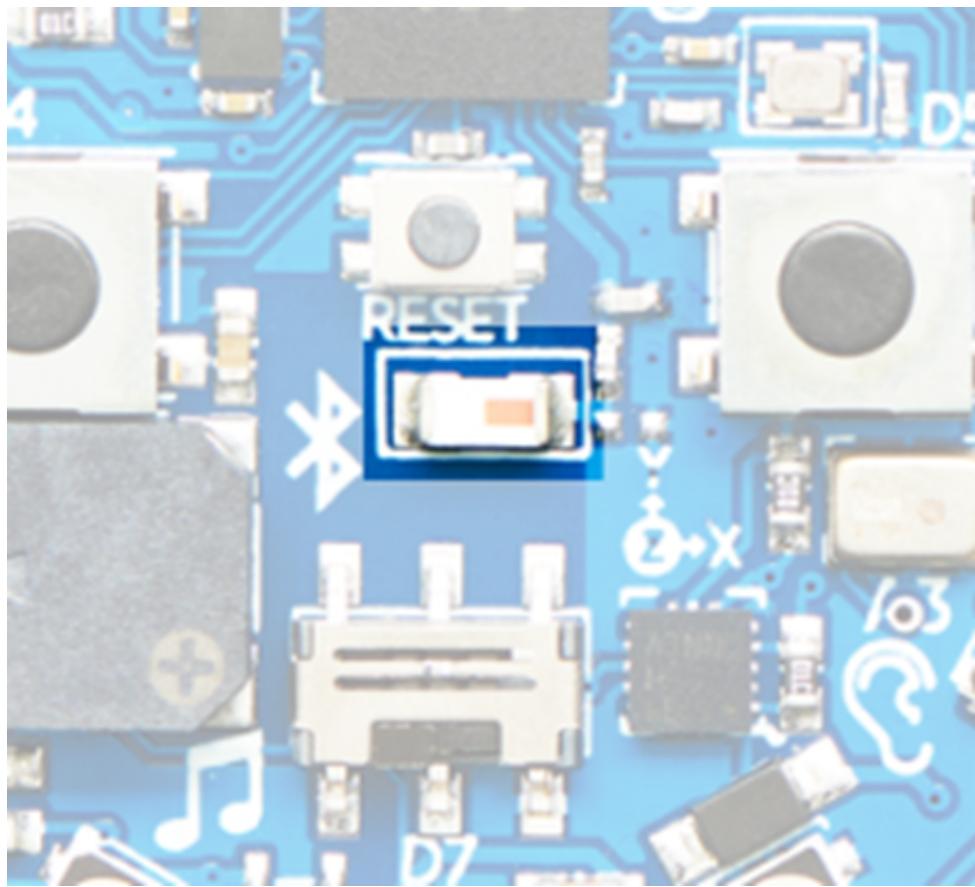


Hình 2.7: Cấu trúc các thành phần trên Circuit PlayGround

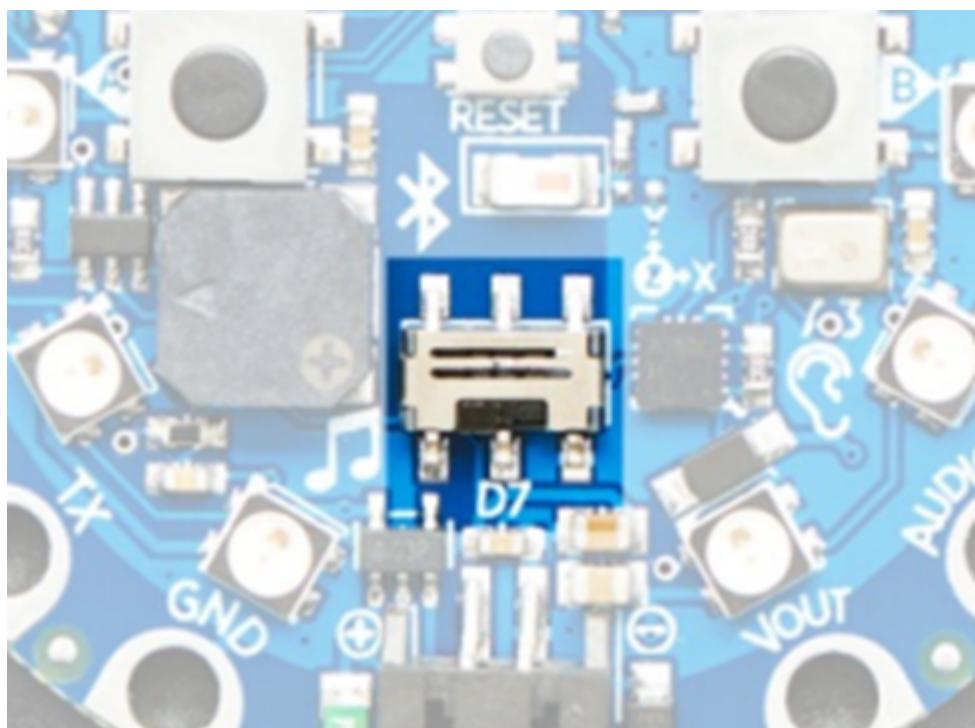
- 1 x Cảm biến nhiệt độ (nhiệt điện trở)
- 1 x Cảm biến ánh sáng (phototransistor). Cũng có thể hoạt động như một cảm biến màu và cảm biến xung
- 1 x Cảm biến âm thanh (micrô MEMS)
- 1 x Loa mini với bộ khuếch đại lớp D (loa từ tính 7,5mm / còi)
- 2 x Nút ấn, có nhãn A và B
- 1 x Công tắc trượt
- 8 x chân đầu vào / đầu ra thân thiện với alligator-clip
- Bao gồm I2C, UART, 6 chân có thể làm đầu vào tương tự, nhiều đầu ra PWM
- Đèn LED "On" màu xanh lục để bạn biết nó được cấp nguồn
- 2 MB dung lượng lưu trữ SPI Flash, được sử dụng chủ yếu với CircuitPython để lưu trữ mã và thư viện.
- Cổng MicroUSB để lập trình và gỡ lỗi
- Cổng USB có thể hoạt động như cổng nối tiếp, bàn phím, chuột, cần điều khiển hoặc MIDI!
- Thiết kế ăng-ten đã được tối ưu và có thêm một chốt cho phép người dùng tắt nguồn của NeoPixels, micrô và cảm biến nhiệt độ / ánh sáng để chỉ có tốc kề hoạt động, để sử dụng năng lượng thấp!
- Đường kính ngoài: 50.6 mm / 2,0 "

### **Năng lượng và dữ liệu:**

- Đầu nối USB Micro B ở trên cùng của bo mạch cho phép giao tiếp nguồn hoặc USB (bộ nạp khởi động tiếp, HID, v.v.).
- Chân cấp nguồn ngoài có thể sử dụng đầu vào DC 6V và có các bảo vệ chống phân cực ngược, quá dòng và nhiệt. Mạch bên trong sẽ sử dụng nguồn điện đầu vào của pin hoặc nguồn điện USB, chuyển từ nguồn này sang nguồn khác một cách an toàn.



Hình 2.8: Vị trí của ăng ten trên bo mạch



Hình 2.9: Vị trí của chân chuyển đổi cấu hình trên bo mạch

Nếu cả hai được kết nối, nó sẽ sử dụng cái nào có điện áp cao hơn. Hoạt động hiệu quả với pin Lithium Polymer hoặc các gói pin 3xAAA của tác giả với đầu nối JST ở cuối. Không có sạc pin tích hợp.

### Ăng ten cho BLE

Ăng-ten Bluetooth nằm ở giữa bo mạch như trên Hình ???. Cần phải đảm bảo không có linh kiện nào gần ăng-ten có thể gây nhiễu, hoặc bề mặt kim loại để tối ưu tầm hoạt động của ăng-ten.

**Chân chuyển đổi cấu hình** Công tắc trượt gần phía dưới trung tâm của Circuit Playground Bluefruit. nó được kết nối với D7 kỹ thuật số, Hình ???. Công tắc sử dụng điện trở kéo để thiết lập cấu hình tiêu thụ năng lượng cao hoặc cấu hình tiêu thụ năng lượng thấp.

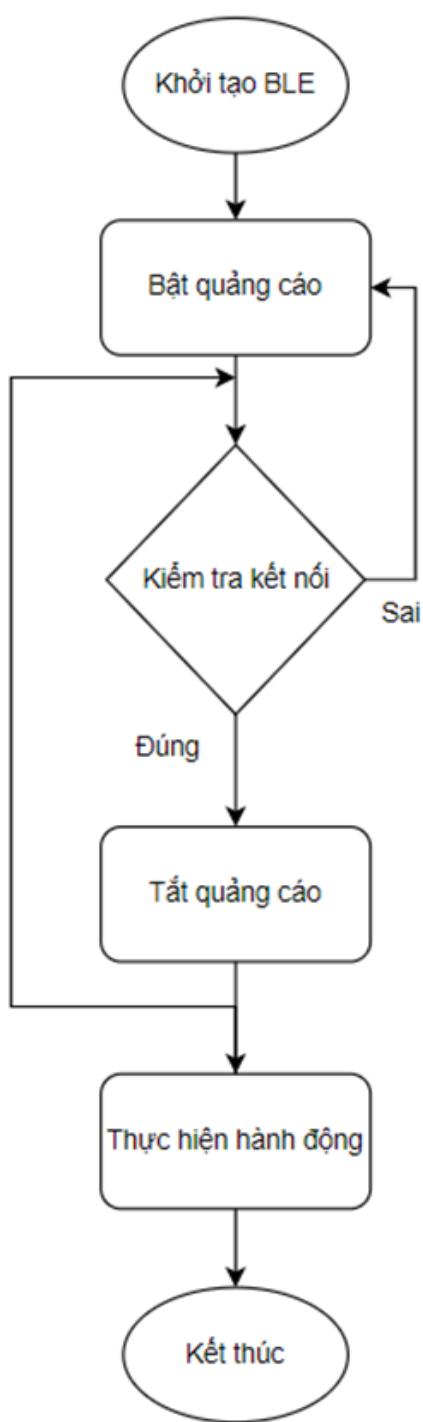
## 2.3. Kỹ thuật thu thập và xử lý tín hiệu

### 2.3.1. Lập trình vi xử lý

Tác giả thực hiện lập trình trên Arduino IDE và thư viện Circuit Playground. Trong phần khởi tạo (setup) các bản tin quảng cáo (advertising) sẽ được khởi tạo và được thực thi. Tiếp theo là cấu hình các hàm kết nối, ngắt kết nối, cấu hình chung GATT Hình ???.

Trong mã nguồn ???, hàm startAdv có nhiệm vụ khởi tạo tên, thêm các dịch vụ và bắt đầu quảng cáo cho thiết bị. Sau đó, cấu hình cho BLE theo định dạng GATT sẽ được thiết lập. Đầu tiên, khởi tạo dịch vụ *positionService* có mã định danh duy nhất(Universally Unique Identifier - UUID) là 0x1821. Tiếp theo, thêm các đặc tính *BLE – GATT – UNIT – ACCELERATION – METRES – PER – SECOND – SQUARED* và *BLE – GATT – UNIT – ANGULAR – ACCELERATION – RADIANS – PER – SECOND – SQUARED* có UUID lần lượt là 0x2713 và 0x2744. Hàm cấu hình *setupPosition* được tác giả sử dụng để cấu hình chi tiết GATT cho BLE. Có 2 service được khởi tạo là giá tốc và con quay hồi chuyển. Tuy nhiên, trong khuôn khổ khóa luận tác giả tập trung vào giá trị giá tốc của cảm biến giá tốc.

**setProperties** - phương thức để thay đổi thuộc tính liên quan đến quyền BLE có thể được đặt thành một hoặc nhiều lựa chọn sau:



Hình 2.10: Lưu đồ hoạt động của thiết bị BLE

---

```
1 void startAdv(void)
2 {
3     // Advertising packet
4     Bluefruit.Advertising.addFlags(BLE_GAP_ADV_FLAGS_LE_ONLY_GENERAL_DISC_MODE);
5     Bluefruit.Advertising.addTxPower();
6
7     // Include HRM Service UUID
8     Bluefruit.Advertising.addService(positionService);
9
10    // Include Name
11    Bluefruit.Advertising.addName();
12
13    /* Start Advertising
14     * - Enable auto advertising if disconnected
15     * - Interval: fast mode = 20 ms, slow mode = 152.5 ms
16     * - Timeout for fast mode is 30 seconds
17     * - Start(timeout) with timeout = 0 will advertise forever (until
18       connected)
19
20     * For recommended advertising interval
21     * https://developer.apple.com/library/content/qa/qa1931/\_index.html
22     */
23
24     Bluefruit.Advertising.restartOnDisconnect(true);
25     Bluefruit.Advertising.setInterval(32, 244); // in unit of 0.625 ms
26     Bluefruit.Advertising.setFastTimeout(30); // number of seconds in
         fast mode
27
28     Bluefruit.Advertising.start(0);           // 0 = Don't stop
         advertising after n seconds
29 }
```

---

Mã nguồn 2.1: Tập lệnh khởi tạo và kết nối Bluetooth từ thư viện của AdaFruit

---

```
1 void setupPosition(void)
2 {
3
4     positionService.begin();
5
6     accelerometerCharacter.setProperties(CHR_PROPS_NOTIFY+CHR_PROPS_READ+CHR_PROPS_WRITE);
7     accelerometerCharacter.setPermission(SECMODE_OPEN,
8             SECMODE_NO_ACCESS);
9     accelerometerCharacter.setFixedLen(9);
10    accelerometerCharacter.setCccdWriteCallback(cccd_callback); // Optionally capture CCCD updates
11
12    accelerometerCharacter.begin();
13    uint8_t accelerometerData[9] = { 0b00000000, 0b00000000,
14        0b00000000,0b00000000,0b00000000,0b00000000,0b00000000,0b00000000,0b00000000}
15    // Set the characteristic to use 8-bit values, with the sensor
16    // connected and detected
17
18    accelerometerCharacter.write(accelerometerData, 9);
19 }
```

---

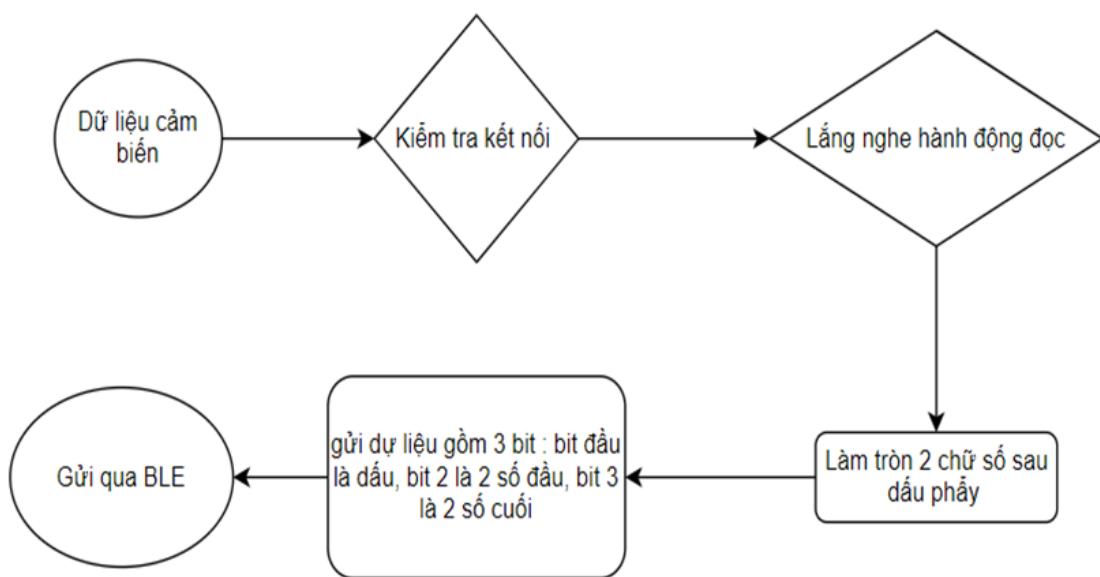
Mã nguồn 2.2: Tập lệnh khởi tạo các thuộc tính GATT

---

```
1 void setupPosition(void)
2 {
3
4     positionService.begin();
5
6     accelerometerCharacter.setProperties(CHR_PROPS_NOTIFY+CHR_PROPS_READ+CHR_PROPS_WRITE);
7     accelerometerCharacter.setPermission(SECMODE_OPEN,
8             SECMODE_NO_ACCESS);
9     accelerometerCharacter.setFixedLen(9);
10    accelerometerCharacter.setCccdWriteCallback(cccd_callback); // Optionally capture CCCD updates
11
12    accelerometerCharacter.begin();
13
14    uint8_t accelerometerData[9] = { 0b00000000, 0b00000000,
15        0b00000000,0b00000000,0b00000000,0b00000000,0b00000000,0b00000000,0b00000000}
16
17    // Set the characteristic to use 8-bit values, with the sensor
18    // connected and detected
19
20    accelerometerCharacter.write(accelerometerData, 9);
21
22
23    gyroscopeCharacter.setProperties(CHR_PROPS_READ);
24    gyroscopeCharacter.setPermission(SECMODE_OPEN, SECMODE_NO_ACCESS);
25    gyroscopeCharacter.setFixedLen(1);
26    gyroscopeCharacter.begin();
27    gyroscopeCharacter.write8(2); // Set the characteristic to 'Wrist'
28            (2)
29 }
```

---

Mã nguồn 2.3: Gửi dữ liệu từ BLE



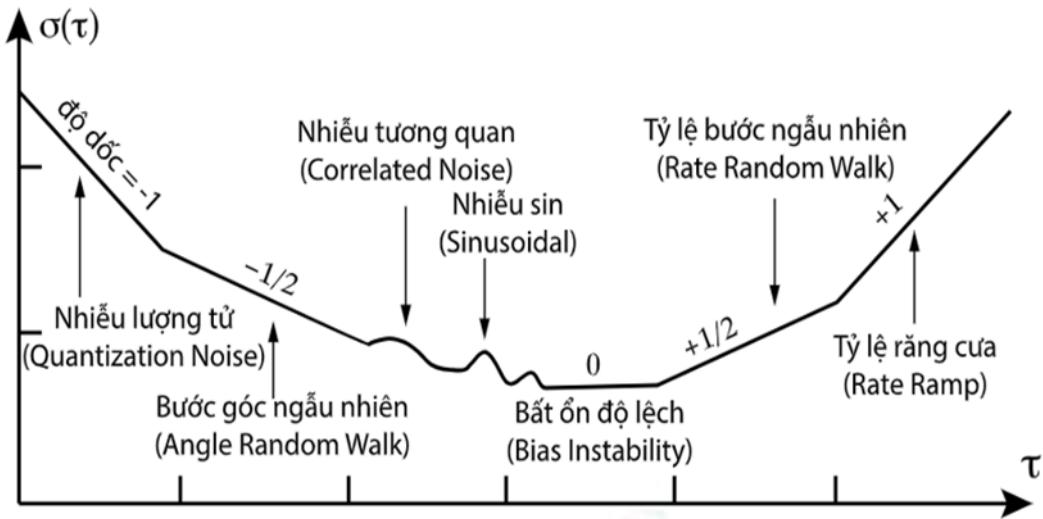
Hình 2.11: Lưu đồ luồng gửi thông tin BLE

- `CHR_PROPS_BROADCAST` = bit (0)
- `CHR_PROPS_READ` = bit (1)
- `CHR_PROPS_WRITE_WO_RESP` = bit (2)
- `CHR_PROPS_WRITE` = bit (3)
- `CHR_PROPS_NOTIFY` = bit (4)
- `CHR_PROPS_INDICATE` = bit (5)

Ngoài ta còn nhiều hàm như `setPermission` để cài đặt tính bảo mật. `setFixedLen` để cấu hình gửi bao nhiêu bit.

### 2.3.2. Hiệu chuẩn cảm biến

Việc thu nhận dữ liệu và tiền xử lý dữ liệu là cần thiết trong các hệ đo. Cảm biến mặc dù đã được hiệu chuẩn tại nơi sản xuất nhưng vẫn cần được hiệu chuẩn trong hệ thống và môi trường đo thực tế. Mục đích của việc hiệu chuẩn là cải thiện hiệu năng của cảm

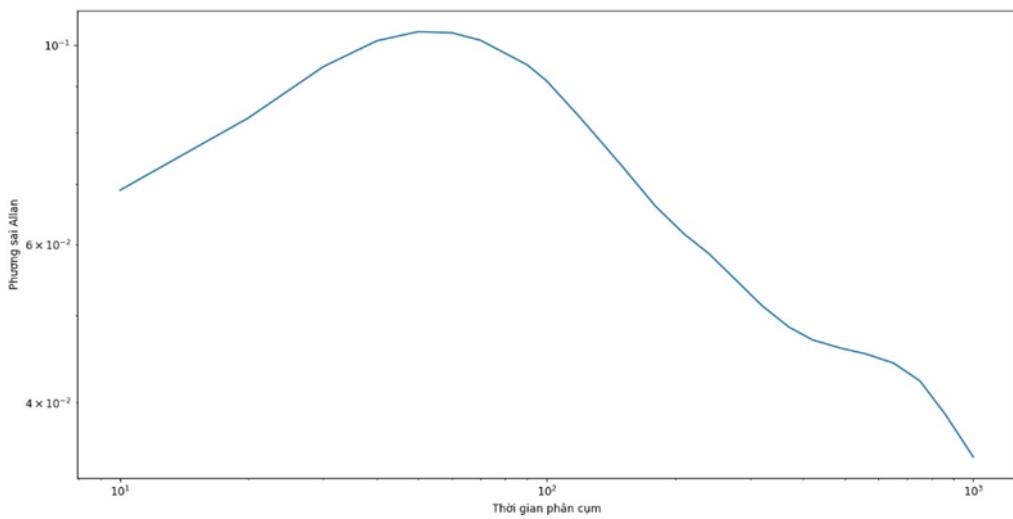


Hình 2.12: Minh họa kết quả phân tích đường cong Allan

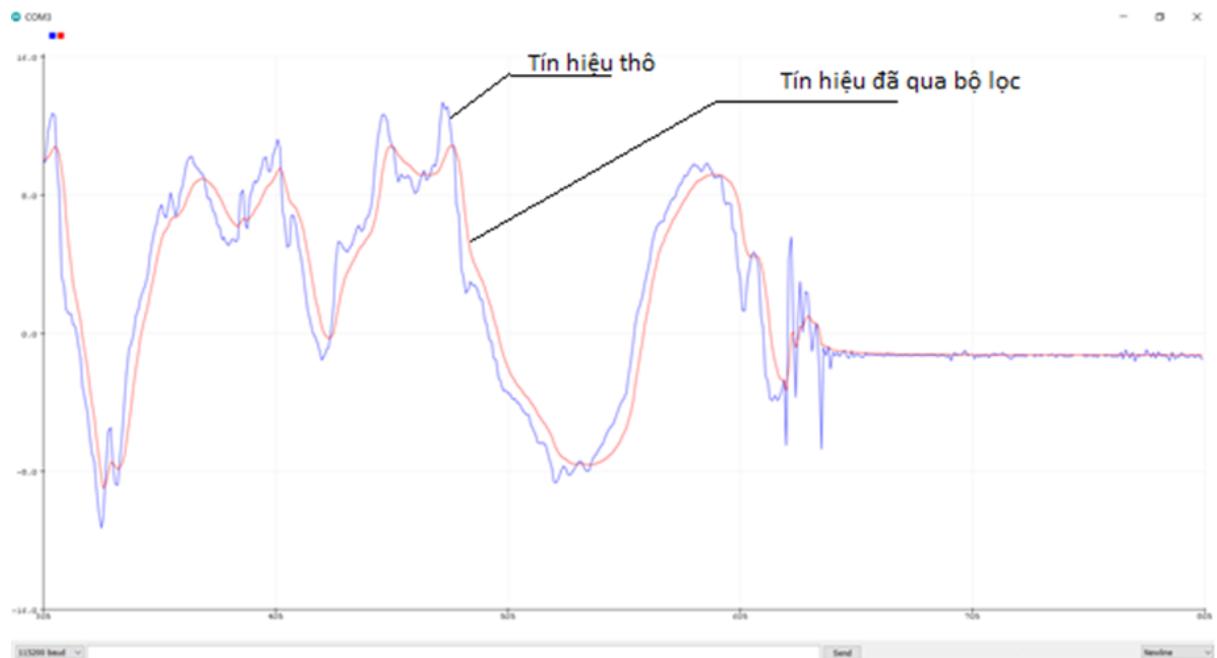
biến bằng cách loại bỏ hoặc giảm thiểu sai số trong dữ liệu đầu ra cảm biến. Lỗi cảm biến phân làm 2 loại i) lỗi mặc định và ii) lỗi ngẫu nhiên. Để xử lý lỗi mặc định, tác giả sử dụng gia tốc trọng trường để hiệu chuẩn cảm biến. Giá trị một trực gia tốc của cảm biến xoay lên trên so với mặt phẳng ngang sẽ nhận giá trị là  $-1g$ , khi xoay xuống dưới nhận giá trị là  $+1g$ . Hiệu chuẩn cảm biến bằng cách xoay cảm biến tạo bởi 6 vị trí tĩnh. Phương pháp xoay cảm biến từ  $+1 g$  đến  $-1 g$  giúp ta thu được giá trị  $0 g$  chính xác và tin cậy.

Lỗi ngẫu nhiên phân tích theo phương sai Allan [**allan**]. Phương sai Allan là phương pháp phân thích chuỗi dữ liệu trong miền thời gian để đo lường sự ổn định tần số trên miền tần số. Phương pháp này là một trong những phương pháp phổ biến nhất hiện nay để xác định và định lượng được nhiều loại nhiễu khác nhau tồn tại trong dữ liệu của cảm biến gia tốc, quán tính, cũng có thể sử dụng phương sai Allan để xác định các nhiễu nội tại trong một hệ thống như là một hàm trung bình cộng của thời gian. Các thành phần nhiễu cảm biến có thể xác định được bằng cách phân tích đồ thị log-log. Độ dốc đường cong Allan sẽ thể hiện các thành phần khác nhau.

Tác giả thu tập được 1211210 mẫu dữ liệu khi để cảm biến đứng với tần số lấy mẫu là 10 Hz ở trong phòng với nhiệt độ bình thường. Kết quả đường cong Allan Hình ?? cho thấy nhiễu lượng tử xuất hiện chủ yếu là nhiễu lượng tử (Quantization noise) dựa trên



Hình 2.13: Biểu đồ phương sai Allan của trục x



Hình 2.14: Kết quả bộ lọc Kalman cho dữ liệu thu được trực tiếp từ trục X của cảm biến gia tốc

việc đổi chiều hệ số góc với Hình ???. Đây là những nhiễu chủ yếu từ bản thân của cảm biến.

Trong khuôn khổ khoá luận, tác giả sử dụng bộ lọc Kalman để lọc các tín hiệu nhiễu đặc biệt là nhiễu lượng tử [**kalman**]. Bộ lọc Kalman là thuật toán sử dụng chuỗi các giá trị đo lường, bị ảnh hưởng bởi nhiễu hoặc sai số, để ước đoán biến số nhằm tăng độ chính xác so với việc sử dụng duy nhất một giá trị đo lường. Bộ lọc Kalman là một bộ lọc đệ quy để ước lượng trạng thái của hệ thống tuyến tính, và cả hệ thống phi tuyến khi áp dụng phép ước lượng phi tuyến sang tuyến tính. Bộ lọc Kalman được ứng dụng rất nhiều trong lĩnh vực kĩ thuật, đặc biệt là lĩnh vực điều khiển. Tín hiệu sau khi được thu thập sẽ được đi qua bộ lọc ở vi xử lý rồi mới được chuyển tới ứng dụng để hiển thị và lưu trữ. Bộ lọc đã được tác giả thực hiện lập trình và có kết quả như Hình ???. Kết quả cho thấy, bộ lọc Kalman đã lược bỏ dữ liệu ít ý nghĩa hoặc nhiễu (ồn) cho một đường biểu diễn mượt mà hơn.

### 2.3.3. Xây dựng phần mềm ứng dụng

Phần mềm ứng dụng được thiết kế với mục tiêu tạo thuận lợi cho người dùng. Nhiệm vụ chính là thực hiện liên kết đến phần cứng và giao diện trực tiếp để người dùng thực hiện các thao tác.

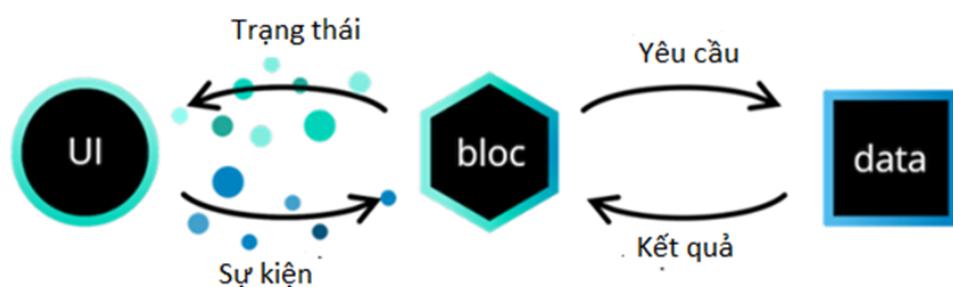
- Ngôn ngữ: Dart
- Framework: Flutter
- Hệ điều hành: Android
- Kết nối với phần cứng: Ble
- Chức năng: Đảm bảo các yêu cầu cơ bản, thuận tiện cho người sử dụng.

Ứng dụng gồm 3 nhóm chức năng chính, Hình ?? bao gồm nhóm chức năng bảo mật, nhóm chức năng chung và nhóm chức năng cá nhân.

**Flutter** là cross-platform dành cho ứng dụng di động viết theo kiểu hướng đối tượng. Nó có thể lập trình cho cả ứng dụng trên nền tảng Android và IOS. Khả năng phát triển nhanh chóng có nhiều thành phần (Widget) có sẵn, đẹp, dễ sử dụng và có nhiều thư viện và cộng đồng sử dụng. Trong ứng dụng này tác giả sử dụng Bloc là một thư viện để quản lý trạng thái cho ứng dụng Flutter B.L.o.C (Business Logic Component). Nhận sự

Nhóm	Chức năng	Mô tả
Bảo mật	Đăng ký	Người dùng đăng ký thông tin và được gửi xác thực qua email
	Đăng nhập	Sử dụng tài khoản mật khẩu đã được đăng ký để đăng nhập vào hệ thống
	Quên mật khẩu	Lấy lại tài khoản của người dùng
Chức năng chung	Kết nối Bluetooth	Kết nối ứng dụng với phần cứng thông qua BLE
	Chọn cảm biến	Lựa chọn cảm biến như (giá tốc , âm thanh , ...)
	Câu hỏi STOPBANG	Trả lời câu hỏi STOP BANG để đánh giá OSA và hiển thị lịch sử các lần trả lời
	Theo dõi thông tin từ cảm biến	Dữ liệu thời gian thực thông qua màn hình giao diện và có đánh giá về tư thế ngủ
Cá nhân	Đổi thông tin cá nhân, mật khẩu	Sửa đổi thông tin cá nhân trên ứng dụng
	Thêm người dùng phụ	Thêm người dùng phụ ví dụ trong cùng 1 gia đình
	Đăng xuất	Thoát khỏi ứng dụng

Hình 2.15: Các chức năng cơ bản của ứng dụng



Hình 2.16: Cấu trúc BLOC

---

```
1 StreamBuilder<List<BluetoothService>>(
2     stream: device.services,
3     initialData: [],
4     builder: (c, snapshot) {
5         if (snapshot.data!.length > 0) {
6             isService = true;
7         }
8         BluetoothService serviceAcclerometer;
9         if (snapshot.data == null || snapshot.data!.length == 0) {
10            return Text("Please contact customer Service");
11        }
12        for (int i = 0; i < snapshot.data!.length; i++) {
13            if (snapshot.data![i].uuid.toString() ==
14                Constants.ACCLEROMETER_SERVICE) {
15                accelerometerService = snapshot.data![i];
16            }
17        }
18        if (accelerometerService == null) {
19            return Text("Please contact customer Service");
20        }
21        for (int i = 0;
22             i < accelerometerService!.characteristics.length;
23             i++) {
24            print(accelerometerService!.characteristics[i].uuid);
25            if (accelerometerService!.characteristics[i].uuid
26                .toString() ==
27                Constants.ACCLEROMETER_CHARACTION) {
28                accelerometerCharactis =
29                accelerometerService!.characteristics[i];
30            }
31        }
32    })));

```

---

Mã nguồn 2.4: Tập lệnh để tìm kiếm dịch vụ cảm biến

---

```
1 void setupPosition(void)
2 {
3
4     positionService.begin();
5
6     accelerometerCharacter.setProperties(CHR_PROPS_NOTIFY+CHR_PROPS_READ+CHR_PROPS_WRITE);
7     accelerometerCharacter.setPermission(SECMODE_OPEN,
8             SECMODE_NO_ACCESS);
9     accelerometerCharacter.setFixedLen(9);
10    accelerometerCharacter.setCccdWriteCallback(cccd_callback); // Optionally capture CCCD updates
11
12    accelerometerCharacter.begin();
13
14    uint8_t accelerometerData[9] = { 0b00000000, 0b00000000,
15        0b00000000, 0b00000000, 0b00000000, 0b00000000, 0b00000000, 0b00000000, 0b00000000 };
16
17    // Set the characteristic to use 8-bit values, with the sensor
18    // connected and detected
19
20    accelerometerCharacter.write(accelerometerData, 9);
21
22
23    gyroscopeCharacter.setProperties(CHR_PROPS_READ);
24    gyroscopeCharacter.setPermission(SECMODE_OPEN, SECMODE_NO_ACCESS);
25    gyroscopeCharacter.setFixedLen(1);
26    gyroscopeCharacter.begin();
27    gyroscopeCharacter.write8(2); // Set the characteristic to 'Wrist'
28            (2)
29 }
```

---

Mã nguồn 2.5: Tập lệnh đọc dữ liệu từ BLE

---

```
1  {
2      "value": "0.88%0.66%0.99@2022-01-01/0.88%0.66%0.99@2022-01-01/0.88%0.66%0.99@2022-01-01"
3      "customer": "62a5f5672ad9c724ef117d76"
4 }
```

---

Mã nguồn 2.6: "Cấu trúc dữ liệu của phần nội dung đẩy lên máy chủ"

kiện như là đầu vào và trả về kết quả là trạng thái. Bloc được xây dựng dựa trên RxDart. Chúng ta có thể chia kiến trúc Flutter thành 3 lớp như Hình ???. Trong đó UI là giao diện của người dùng, nơi người dùng nhìn thấy dữ liệu, thao tác. Bloc là thư viện để quản lý các trạng thái của ứng dụng. Các sự kiện là các hành động như ấn nút, nhập dữ liệu, v.v.

Khi đã lấy được đối tượng tính năng (charactis instance), tác giả sẽ liên tục đẩy hành động đọc (read) từ phần mềm ứng dụng đến vi điều khiển. Khi vi điều khiển nhận được hành động đọc được gửi từ ứng dụng sẽ trả lại tín hiệu của cảm biến. Ở đây, các BLE gửi các dữ liệu theo định dạng là mảng các phần tử có kiểu là UInt8. Do đó, sau khi nhận được dữ liệu từ phần cứng, dữ liệu dạng UInt8 được chuyển về dạng Number. Với quy trình như vậy, hệ thống có độ trễ rất thấp (real time). Cuối cùng việc lưu trữ thông tin cảm biến được thông qua giao thức Giao thức Truyền tải Siêu Văn Bản (Hyper Text Transfer Protocol - HTTP) truyền lên phía máy chủ(backend) và được thực hiện ở đoạn Mã ???. Việc đẩy tín hiệu như vậy, tránh được tình trạng tắc nghẽn ở phía backend. Trong thời gian tới, cơ chế nghe/nhận (pub/sub) sẽ được áp dụng để tránh tình trạng gọi các yêu cầu (request) liên tục. Dữ liệu sẽ được đẩy lên theo định dạng Mã ???:

Ngoài ra, tác giả phát triển thêm tính năng bộ câu hỏi STOPBANG nhằm đánh giá ban đầu người dùng. Và những dữ liệu này có trọng số quan trọng trong việc đánh giá chỉ số AHI sử dụng học máy sau này. Tính năng thêm người dùng được sử dụng để tránh phải thao tác tạo mới nhiều lần nhưng vẫn giữ được tính bảo mật, lấy lại dữ liệu khi mà quên tài khoản, mật khẩu.

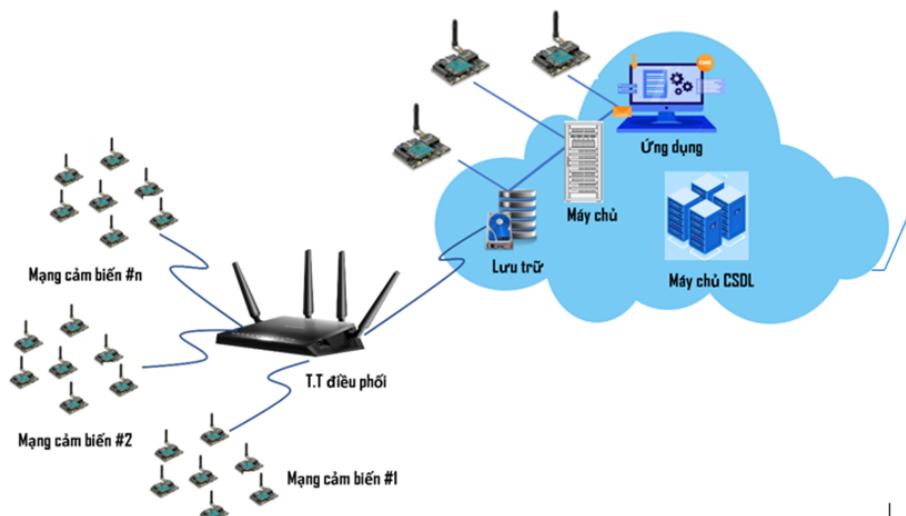
#### 2.3.4. Thiết kế và xây dựng hệ thống lưu trữ

Dữ liệu (data) là cơ sở của công nghệ trí tuệ nhân tạo để phân tích, đánh giá, dự đoán nhiều lĩnh vực trong cuộc sống. Vì mỗi thiết bị phần cứng chỉ có không gian bộ nhớ có

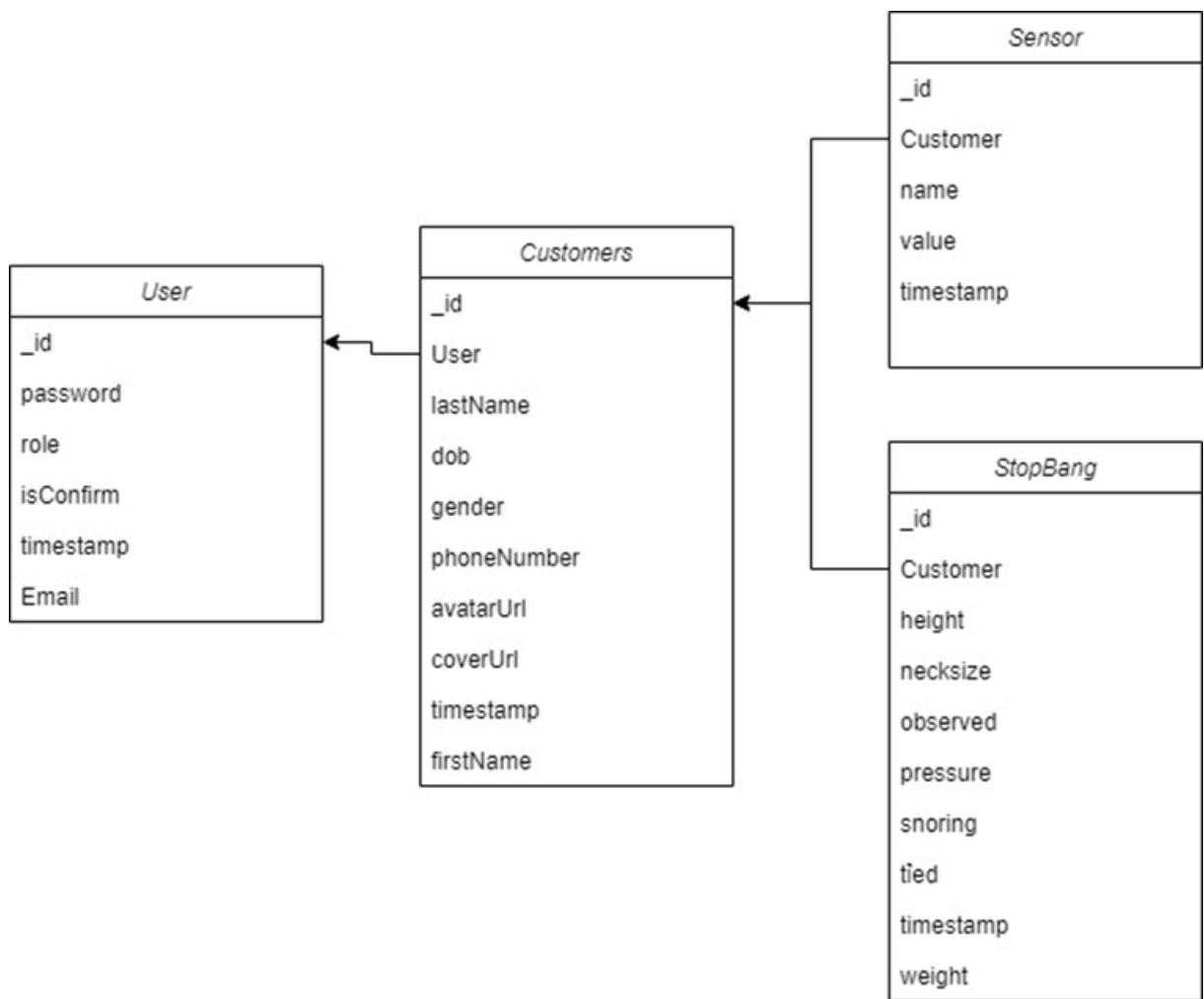
hạn nên việc lưu trữ trên hệ thống đám mây (cloud) đang là phương pháp phổ biến nhất. Hình ???. Việc triển khai dữ liệu lên cloud giúp xóa bỏ khoảng cách địa lý, thuận tiện cho việc trích xuất bất kỳ nơi nào miễn là có kết nối internet, qua đó có thể dễ dàng xuất ra các file text, csv và thuận tiện chia sẻ trao đổi với các nhóm nghiên cứu khác. Mục tiêu dài hạn của hướng nghiên cứu này là việc sử dụng cơ sở dữ liệu đủ lớn cho các mô hình toán học nhằm dự đoán, hỗ trợ ra quyết định trong các hoạt động sàng lọc và chẩn đoán chứng OSA ở người.

- MongoDB được thiết kế để có thể truy vấn hiệu quả các bộ dữ liệu rất lớn, ngay cả khi dữ liệu được phân chia trên nhiều máy chủ, miễn là có thể chọn share key đoạn phù hợp khớp với các truy vấn đọc phổ biến nhất của mình.
- Hỗ trợ tốt đối với dữ liệu TimeStamp.
- Các chỉ mục được phân nhánh tối ưu hóa hiệu suất và lưu trữ chỉ mục.
- Tự động xóa dữ liệu cũ hơn. Lưu trữ dữ liệu bằng kho lưu trữ trực tuyến Atlas và dễ mở rộng

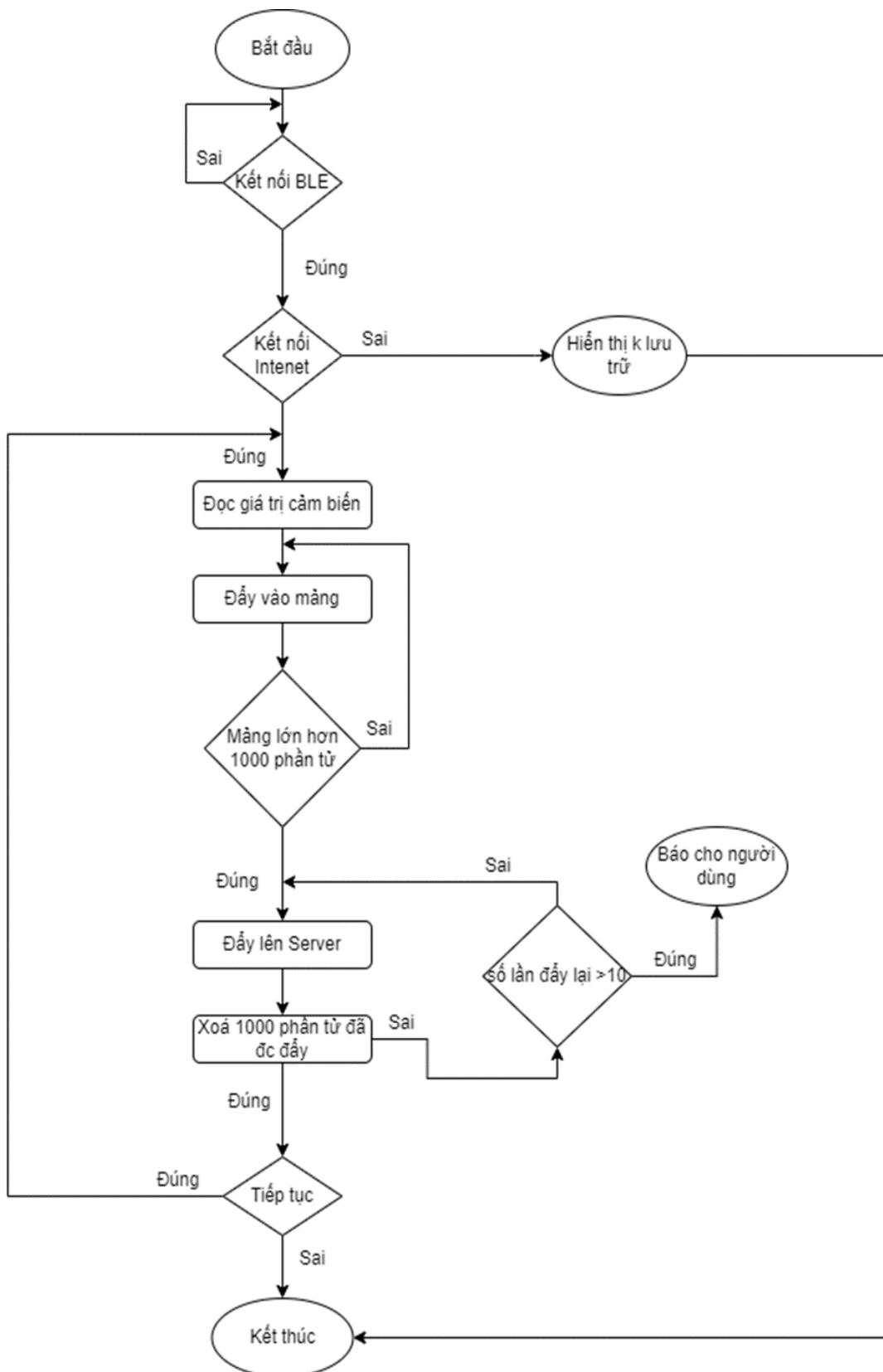
Trong khóa luận này, phần máy chủ của hệ thống được xây dựng trên nền tảng NodeJs và triển khai trên AWS - nền tảng đám mây cho phép các lập trình viên xây dựng, triển khai, quản lý và mở rộng ứng dụng. Cơ sở dữ liệu sử dụng trên nền tảng MongoDb Atlas hỗ trợ free 500Mb.



Hình 2.17: Mô hình tích hợp giữa mạng cảm biến và cấu trúc dữ liệu đám mây



Hình 2.18: Cấu trúc lưu trữ dữ liệu



Hình 2.19: Lưu đồ thuật toán lưu trữ dữ liệu cảm biến

Để tránh tình trạng quá tải server khi có quá nhiều yêu cầu truy cập cùng lúc đến, tác giả đã gộp các giá trị cảm biến vào 1 mảng 1000 phần tử rồi đẩy mảng đó lên backend. Việc lưu trữ này yêu cầu phải xác định được chính xác thời gian của từng giá trị của biến số trên ứng dụng di động và truyền lên cùng giá trị của 3 trục x, y, z. Việc lưu trữ dữ liệu được tác giả mô tả như lược đồ, Hình ?? bao gồm 2 trường hợp:

- Khi người dùng không có kết nối mạng thì người dùng chỉ được phép truy cập vào BLE và hiển thị dữ liệu không có lưu trữ.
- Khi trường hợp người dùng đã đăng nhập và có kết nối mạng thì ứng dụng sẽ tự động lưu trữ khi thu thập đủ 1000 giá trị. Nếu không thành công 10 lần sẽ báo cho người dùng.

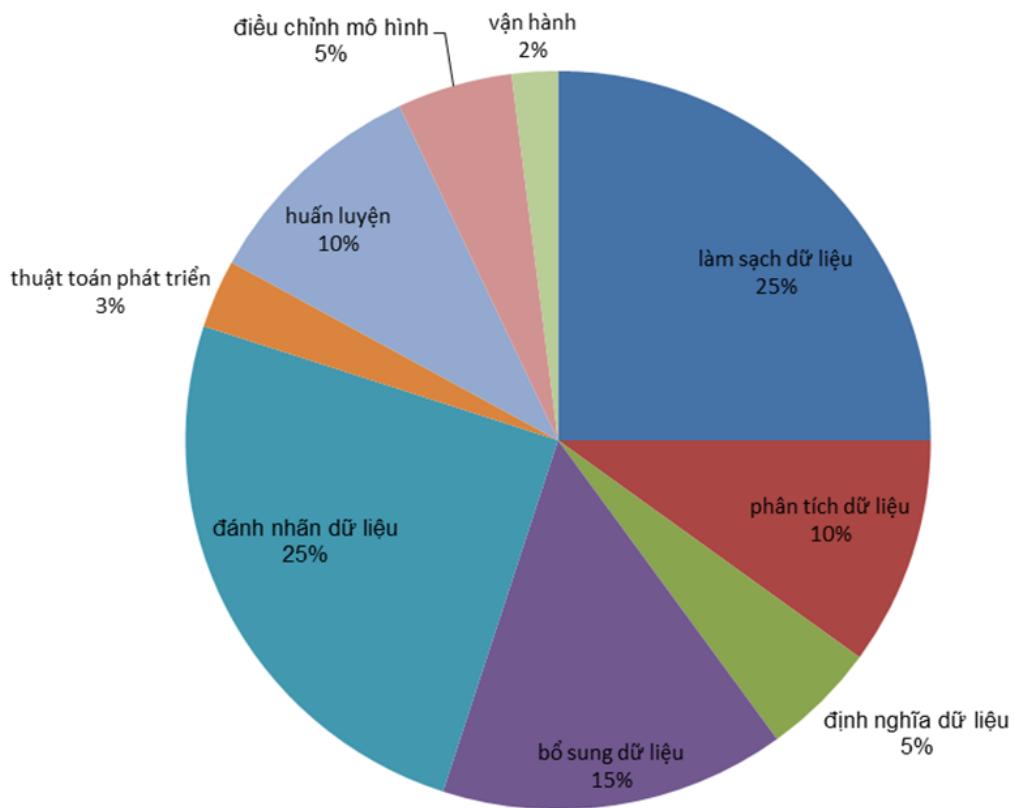
### 2.3.5. Tìm hiểu, ứng dụng phân loại tư thế ngủ bằng học máy

Tác giả cũng đã tìm hiểu nhiều mô hình, phương pháp để phân loại các tư thế ngủ, tư thế cơ bản của con người và đánh giá chỉ số AHI dự trên các tín hiệu cảm biến thu được. Các bước cơ bản để tiến hành dự án học máy liên quan đến các tín hiệu cảm biến:

- Thu thập dữ liệu (bao gồm thu thập và gắn nhãn cho dữ liệu)
- Khám phá dữ liệu (đánh giá cân bằng dữ liệu, tỉ lệ dữ liệu có ý nghĩa)
- Chuẩn bị dữ liệu (làm sạch dữ liệu, tạo ra các đặc tính trên miền thời gian và miền tần số)
- Mô hình hóa dữ liệu (lựa chọn ra các mô hình phù hợp)
- Lựa chọn tính năng (lựa chọn ra các tính năng có ý nghĩa cao đối với mô hình)
- Tinh chỉnh mô hình

Jeng PY và cộng sự đã đề xuất chế tạo 2 thiết bị đeo ở cổ và ở cổ tay để đánh giá tư thế ngủ ở người [**Jeng**]. Trong dự án này, tín hiệu thu được ở thiết bị đeo ở tay được chia thành những cửa sổ 1 giây rồi trích xuất các tính năng trên cửa sổ đó. Cảm biến đeo ở cổ sẽ được sử dụng để lấy nhãn tín hiệu theo phương pháp lấy đa số của tín hiệu trong cửa sổ. Nhóm tác giả đã sử dụng mô hình SVM và RF để đánh giá và đạt được kết quả có độ

chính xác lần lượt là 82% và 72%. Nhóm của Saha S., Kabir M và cộng sự đã tiến hành nghiên cứu 1 thiết bị đeo được sử dụng bao gồm cảm biến gia tốc, cảm biến âm trên 31 đối tượng thử nghiệm. Sau đó họ tiến hành loại bỏ các bộ dữ liệu có độ dài dưới 2 giờ và cuối cùng sử dụng so sánh ngưỡng để xác định chứng OSA bằng việc phân chia các cửa sổ 10s với độ lặp 80% [Saha]. Trong khi đó, nhóm của Syeda Zuriat-e-Zehra Ali và cộng sự đã nghiên cứu và thử nghiệm thiết bị gối ngủ để tự điều chỉnh hoặc báo hiệu khi có chứng ngưng thở khi ngủ dựa trên các tín hiệu thô như nồng độ Oxi trong máu, nhịp tim [27]. Jarvis L, Moninger S và cộng sự đã trình bày hệ thống phát hiện đánh giá 5 tư thế gồm nằm, nằm tựa, ngồi thẳng, đứng, đi bộ với tập dữ liệu được lấy từ 2 cảm biến gắn ở cổ và đùi [Syeda]. Dữ liệu được lấy mẫu với tần số 25 Hz sau đó được lưu vào bộ nhớ cục bộ trên điện thoại rồi gửi bản csv qua mail. Mô hình học máy gồm hồi quy logistic, SVM, DT với độ chính xác cao  $> 96\%$  đã được sử dụng đánh giá tập dữ liệu gồm 6 hành động thường ngày của con người như đứng, ngồi, đi bộ, lên cầu thang, xuống cầu thang và nằm [Uday]. Nhóm nghiên cứu của Gomes E, Bertini L và cộng sự đã nghiên



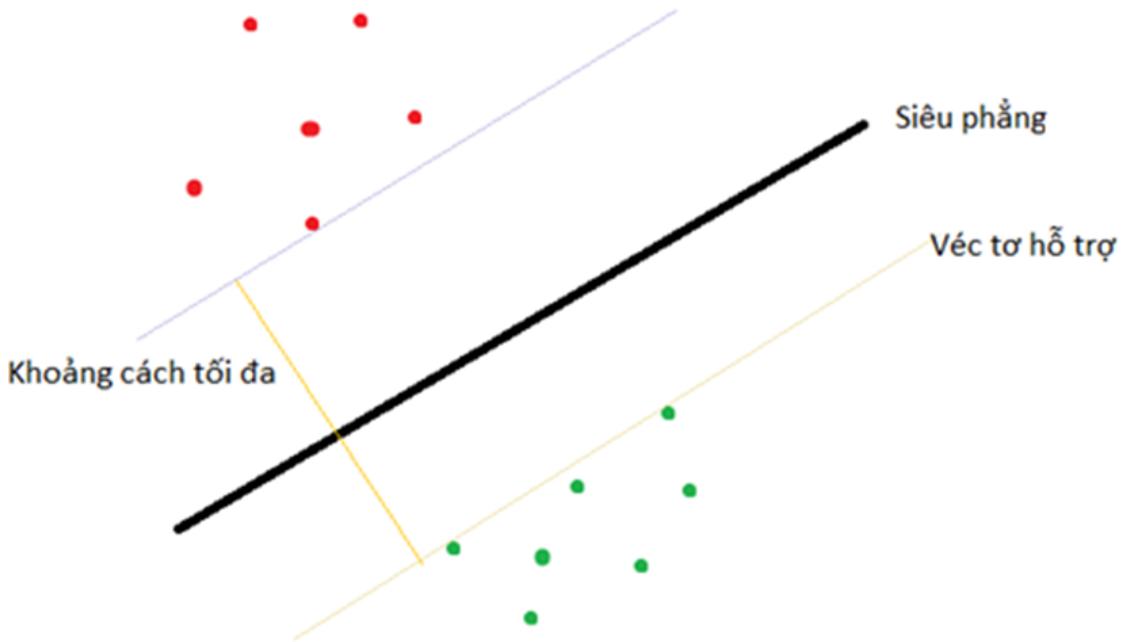
Hình 2.20: Phân bố thời gian sử dụng đối với dự án học máy

cứu, xây dựng, đánh giá giữa 3 mô hình: K-Nearest-Neighbor (KNN), cây quyết định (Decision tree) và SVM. Trong các bước tiền xử lý các tác giả đã phân đoạn dữ liệu theo cửa sổ 2.5s không che phủ sau đó phân tích, chuẩn hoá dữ liệu và đã có độ chính xác > 97% đối với việc phát hiện tư thế. Ở Việt Nam, nhóm tác giả Vũ Ngọc Thanh Sang và Nguyễn Đức Thắng đã phát triển thiết bị thu thập dữ liệu từ điện thoại sau đó qua các bước xử lý dữ liệu, trích xuất tính năng và phân loại bằng các mô hình K-hàng xóm gần nhất (KNN) với độ chính xác là 100% với toàn bộ tư thế ngoại trừ lái xe là 80% [Sang]. Qua tổng quan tài liệu tác giả nhận thấy các phương pháp học máy cổ điển đang chiến ưu thế hơn so với các phương pháp học sâu vì phát triển nhanh và dễ dàng và phù hợp với tính chất của bài toán đánh giá các tư thế của con người sử dụng cảm biến gia tốc. Trong đó, nổi bật lên là mô hình SVM, hồi quy Logistic và Random Forest. Từ đó, tác giả sẽ tập trung tìm hiểu và hướng tới áp dụng cho tập dữ liệu của tác giả.

**Hồi quy logistic - LR:** Đây là phương thức tốt nhất cho các vấn đề phân loại nhị phân (vấn đề với hai lớp giá trị). Hồi quy logistic giống như hồi quy tuyến tính với mục đích là để tìm ra các giá trị cho các hệ số mà trọng lượng mỗi biến đầu vào. Không giống như hồi quy tuyến tính, dự đoán đầu ra được chuyển đổi bằng cách sử dụng một hàm không tuyến tính được gọi là hàm logistic. Hàm logistic trông giống như một chữ S lớn và sẽ biến đổi bất kỳ giá trị nào thành 0-1. Tuy nhiên, nhược điểm của nó là chỉ giải quyết được bài toán phân loại 2 lớp. Để giải quyết được những bài toán đa lớp chúng ta có thể sử dụng mô hình Softmax Logistic là dạng tổng quát của hồi quy Logistic.

**Máy vec tơ hỗ trợ (Support vector machines - SVM):** xây dựng một mặt siêu phẳng được sử dụng để phân chia không gian biến đầu vào. Trong SVM, một mặt siêu phẳng được chọn để phân tách tốt nhất các điểm trong không gian các biến đầu vào theo lớp của chúng, hoặc là lớp 0 hoặc lớp 1. Trong không gian hai chiều, có thể hình dung nó như một đường thẳng và giả sử rằng tất cả các biến đầu vào có thể được tách hoàn toàn bằng đường thẳng này. Thuật toán SVM tìm ra các hệ số dẫn đến sự phân tách tốt nhất của các lớp theo mặt siêu phẳng Hình ??.

Khoảng cách giữa mặt siêu phẳng và điểm dữ liệu gần nhất được gọi là biên. Mặt siêu phẳng tốt nhất hoặc tối ưu có thể tách riêng hai lớp là dòng có biên lớn nhất. Chỉ những điểm này có liên quan đến việc xác định hyperplane và trong việc xây dựng các điểm phân loại. Những điểm này được gọi là các vector hỗ trợ. Chúng hỗ trợ hoặc xác định hyperplane. Trong thực tế, một thuật toán tối ưu được sử dụng để tìm các giá trị cho

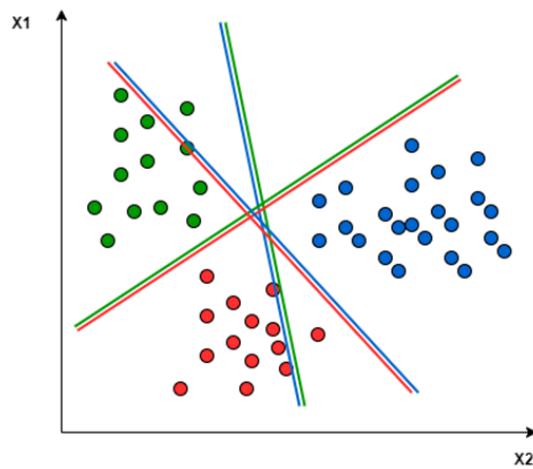


Hình 2.21: Tối ưu siêu phẳng sử dụng thuật toán SVM

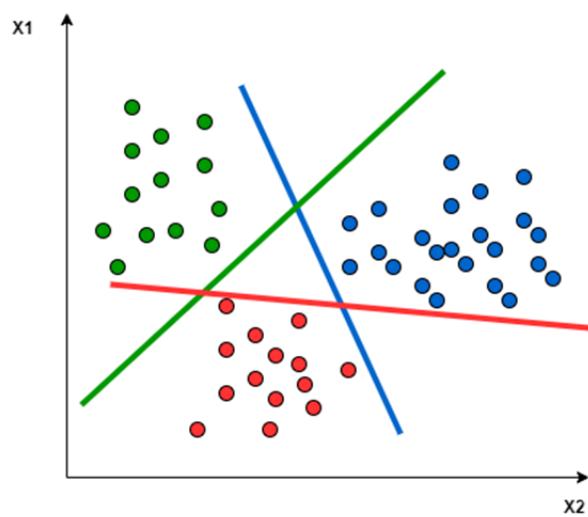
các hệ số tối đa hóa biên. SVM có thể là một trong những phương pháp phân loại hàng đầu mạnh mẽ nhất và đáng thử trên tập dữ liệu. Cũng như hồi quy Logistic thì SVM cũng chỉ sử dụng để phân loại nhị phân. Để giải quyết vấn đề này thì có 2 phương pháp:

- Một với một (one vs one): Một mặt siêu phẳng được thiết lập để phân tách giữa hai lớp, bỏ qua các điểm của lớp thứ ba. Điều này có nghĩa là sự phân tách chỉ tính đến điểm của hai lớp trong sự phân tách hiện tại. Ví dụ: đường màu đỏ-xanh dương sẽ tách tối đa khoảng cách chỉ giữa các điểm màu xanh lam và màu đỏ Hình ??.
- Một với nhiều (one vs rest): Cần một mặt siêu phẳng để tách biệt giữa một lớp và tất cả các lớp khác cùng một lúc. Điều này có nghĩa là sự tách biệt có tính đến tất cả các điểm, chia chúng thành hai nhóm; một nhóm cho các điểm của lớp và một nhóm cho tất cả các điểm khác. Ví dụ: đường màu sẽ tách tối đa hóa khoảng cách giữa các điểm màu lục và tất cả các điểm khác cùng một lúc Hình ??.

**Rừng ngẫu nhiên (Random Forest - RF):** được xây dựng trên cơ sở thuật toán Decision Tree (cây quyết định). Mỗi cây quyết định sẽ khác nhau (có yếu tố ngẫu nhiên khác nhau). Sau đó kết quả dự đoán được tổng hợp từ các cây quyết định. Trong thuật



Hình 2.22: Thuật toán một với một



Hình 2.23: Thuật toán một với nhiều

toán Decision Tree, khi xây dựng cây quyết định nếu để độ sâu tùy ý thì cây sẽ phân loại đúng hết các dữ liệu trong tập training dẫn đến mô hình có thể dự đoán tệ trên tập validation/test, khi đó mô hình bị quá khớp (overfitting). Thuật toán Random Forest gồm nhiều cây quyết định, mỗi cây quyết định đều có những yếu tố ngẫu nhiên: Lấy ngẫu nhiên dữ liệu để xây dựng cây quyết định. Lấy ngẫu nhiên các thuộc tính để xây dựng cây quyết định. Do mỗi cây quyết định trong thuật toán Random Forest không dùng tất cả dữ liệu training, cũng như không dùng tất cả các thuộc tính của dữ liệu để xây dựng cây nên mỗi cây có thể sẽ dự đoán không tốt, khi đó mỗi mô hình cây quyết định không bị overfitting mà có thể bị underfitting, hay nói cách khác là mô hình có high bias. Tuy nhiên, kết quả cuối cùng của thuật toán Random Forest lại tổng hợp từ nhiều cây quyết định, thế nên thông tin từ các cây sẽ bổ sung thông tin cho nhau, dẫn đến mô hình có low bias và low variance, hay mô hình có kết quả dự đoán tốt. Những kiến thức cơ bản về học máy sẽ được ứng dụng sâu trong những nghiên cứu tới đây của tác giả.

## **Chương 3**

# **KẾT QUẢ THỰC NGHIỆM VÀ ĐÁNH GIÁ**

### **3.1. Đánh giá tư thế ngủ sử dụng cảm biến gia tốc**

Tác giả tiến hành thử nghiệm bằng cách gắn bộ kit tại phần xương ức ở cổ khi ngủ. Phòng ngủ kín với nhiệt độ ổn định kín. Đầu tiên, thiết bị sẽ được gắn lên vị trí đo bằng băng keo y tế. Khởi động thiết bị, sau đó tiến hành đăng nhập trên ứng dụng trên điện thoại. Cuối cùng kết nối với BLE với thiết bị phần cứng có tên là “Adafruit Playground” và chọn dịch vụ cảm biến gia tốc. Tác giả thử nghiệm theo từng tư thế ngủ bao gồm: nằm ngửa, nằm nghiêng trái, nằm sấp, nằm nghiêng phải. Mỗi tư thế sẽ được thử nghiệm trong 5 phút và cho ra kết quả tương đối chính xác. Khi chuyển các tư thế nằm ngủ, sẽ có 2 trường hợp xảy ra:

- Nếu chuyển động thay đổi tư thế nhẹ nhàng, kết quả vẫn có độ chính xác cao.
- Chuyển động nhanh có thể xảy ra và khó phát hiện đó là tư thế ngủ nào.

Trường hợp nằm ngủ với góc nghiêng (so với mặt phẳng ngang) lớn hơn 450, không được ghi nhận là tư thế ngủ. Tác giả xác nhận dữ liệu lấy ra từ cảm biến, dữ liệu truyền qua BLE, dữ liệu được lưu trữ trên đám mây đồng bộ với nhau, không có sự mất mát.

Các bước tiến hành thực nghiệm như Hình ??:

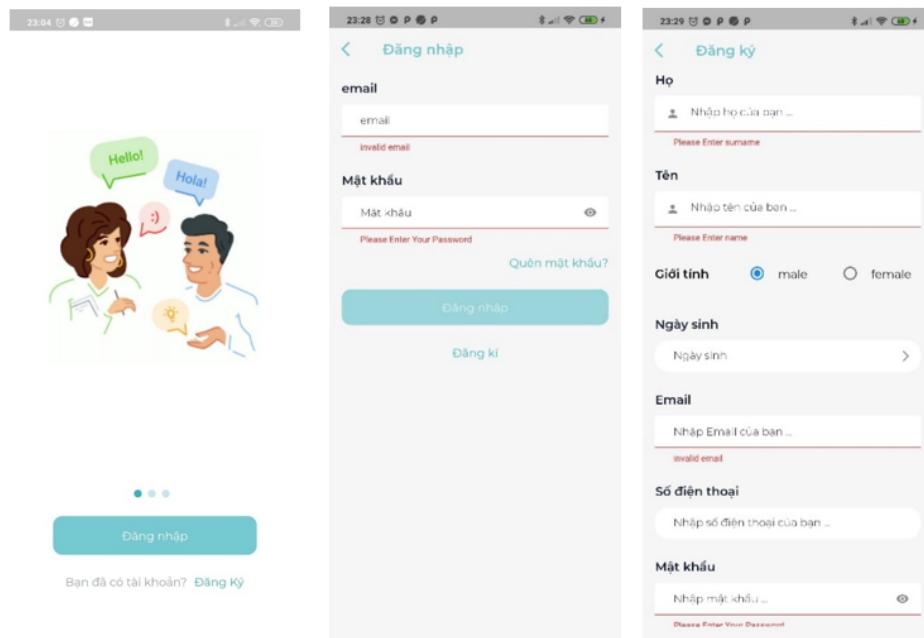
- Gắn cỗ định kit bằng băng ý tế lên phần xương ức ở cổ.
- Tiến hành đăng nhập.
- Lựa chọn thiết bị BLE.
- Kiểm tra kết quả
- Tiếp cận cơ bản mô hình học máy.

Một trong những nhiệm vụ mà tác giả được giao khi thực hiện khóa luận là phát triển một phần mềm ứng dụng cho phép người dùng sử dụng một cách đơn giản. Với nhận thức của tác giả, dựa trên góp ý của nhóm nghiên cứu, phần mềm ứng dụng phải đáp ứng những yêu cầu như dễ dàng thao tác, giao diện thân thiện, tính năng cơ bản.

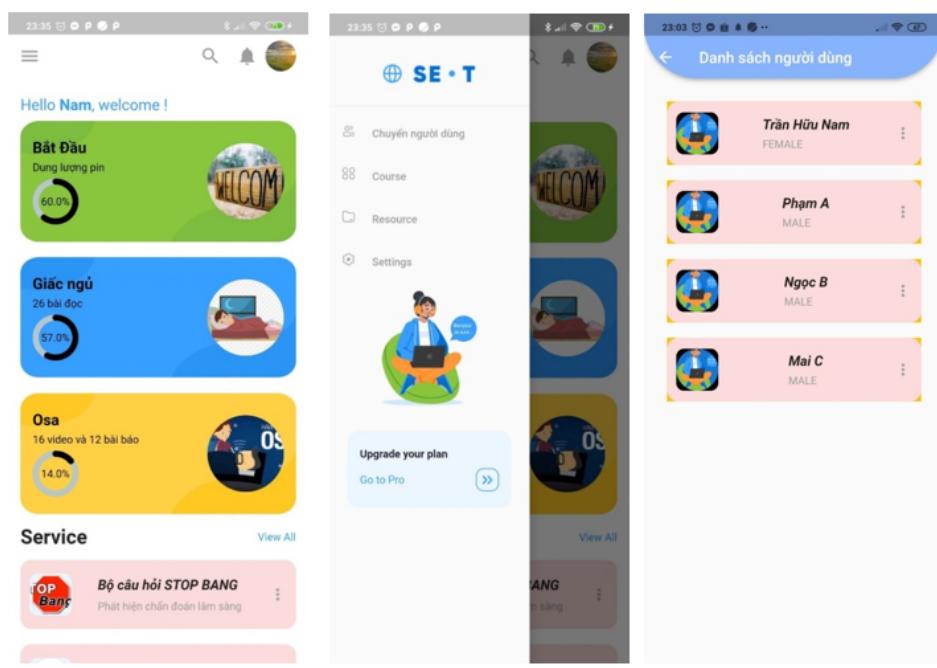
Sau khi cài đặt, cửa sổ đăng nhập (cho người đã đăng ký) hoặc đăng ký mới có giao diện như trong Hình ???. Với người mới bắt đầu sử dụng và muốn lưu trữ dữ liệu cần phải đăng ký tài khoản và xác thực qua email. Mục đích để xác định người đang sử dụng và lấy lại mật khẩu khi cần thiết.



Hình 3.1: Thực nghiệm với kit Adafruit Playground



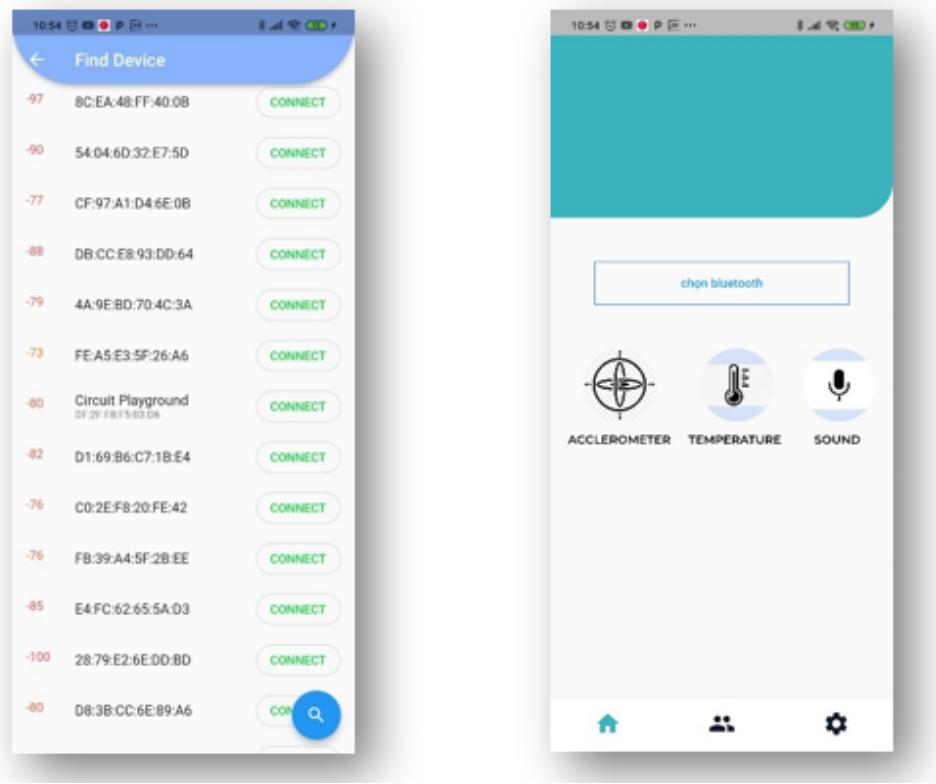
Hình 3.2: Các chức năng đăng ký, đăng nhập



Hình 3.3: Giao diện trang chủ

Hình ?? là giao diện khi người dùng đăng nhập thành công, bao gồm các tính năng: Kết nối BLE và đọc dữ liệu, chuyển người dùng, xem thông tin người dùng v.v.

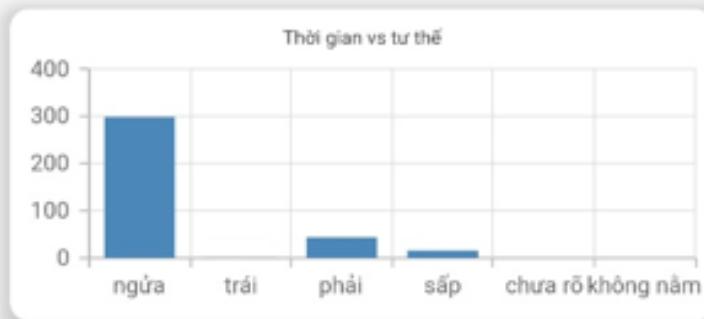
Hình ?? là giao diện kết thể hiện các thiết bị đang mở trạng thái quảng cáo. Sau



Hình 3.4: Kết nối với BLE và hiển thị dữ liệu

khi chọn đúng thiết bị phần cứng thì giao diện sẽ chuyển tới màn hình chọn dịch vụ. Sau khi chọn dịch vụ cảm biến màn hình tự động chuyển đến màn hình hiển thị các biểu đồ. Trong khuôn khổ khoá luận, tác giả sử dụng so sánh ngưỡng dữ liệu thu được từ 3 trực của cảm biến gia tốc thu được với các giá trị đặt sẵn sau nhiều lần thực nghiệm Mã nguồn ??.

Hình ?? là phần quan trọng nhất của ứng dụng. Phần đầu tiên là biểu đồ giá trị 3 trực theo thời gian thực. Phần 2 là tổng thời gian theo tư thế ngủ tính theo phút. Và cuối cùng là tư thế hiện tại của người dùng. Tuy nhiên việc so sánh ngưỡng như vậy chưa mang tính thống kê trong khuôn khổ khoá luận. Vì vậy, các mô hình học máy cần phải được sử dụng trong tương lai để đánh giá và kết luận tư thế ngủ với độ chính xác cao. Theo đó, tư thế ngủ liên tục được cập nhật và tổng hợp sau mỗi 10 giây.



Tư thế hiện tại của bạn là :

**Ngửa**

Hình 3.5: Màn hình hiển thị giá trị của 3 trục cảm biến gia tốc

---

```
1 static Function getPositionSleep = (double x, double y, double z) {  
2     if ((-6.5 < y && y < 6.5)) {  
3         if (-7.07 < x && x < 7.07) {  
4             if (z > 0) {  
5                 return 1; // ngua  
6             }  
7             if (z < 0) {  
8                 return 4; //sap  
9             }  
10        }  
11        if (x > 3) return 2; //trai  
12        if (x < -3) return 3; //phai  
13    }  
14    return 6; // khong phai nam  
15};
```

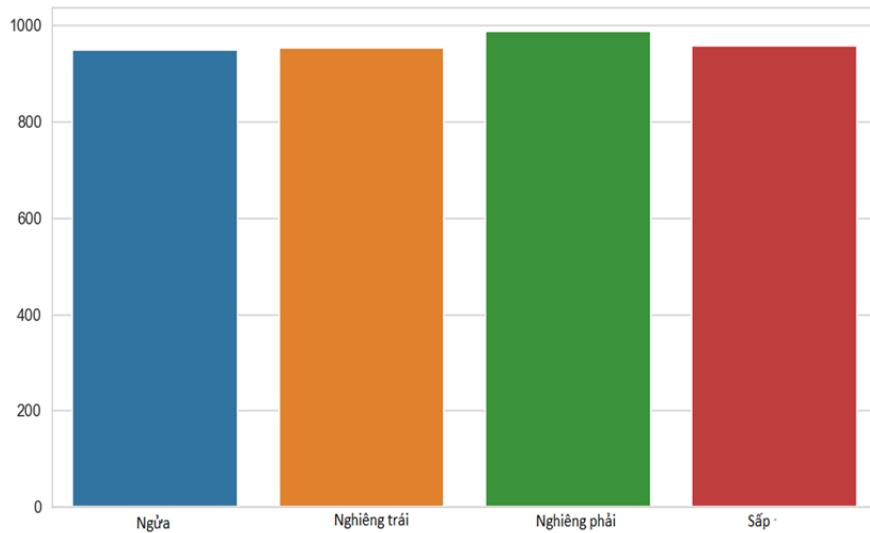
---

Mã nguồn 3.1: Tập lệnh đánh giá tư thế ngủ bằng ngưỡng

### 3.2. Học máy với dữ liệu đánh giá từ thiết bị

Sau khi tìm hiểu những bước cơ bản từ chương 2 tác giả đã có những bước tiến hành như sau:

- Thu thập dữ liệu và đánh nhãn: Tác giả thu tập 3859 mẫu dữ liệu theo từng tư thế ngủ: ngửa, nghiêng trái, nghiêng phải và sắp với 1 đối tượng trong nhóm. Mỗi tư thế tác giả thực hiện trong 100 giây là dừng lại để đánh nhãn trực tiếp sau mỗi lần thử nghiệm. Việc lấy mẫu như vậy có tác dụng i) cân bằng dữ liệu ii)) tối ưu độ chính xác của nhãn.
- Phân tích dữ liệu: Xóa bỏ những dữ liệu ban đầu của mỗi hoạt động. Trích xuất tính năng theo miền thời gian (trung bình, phương sai, giá trị nhỏ nhất, giá trị lớn nhất,...) theo các cửa sổ 1 giây.
- Đánh giá tương quan và loại bỏ tính năng.
- Sử dụng các mô hình học máy.



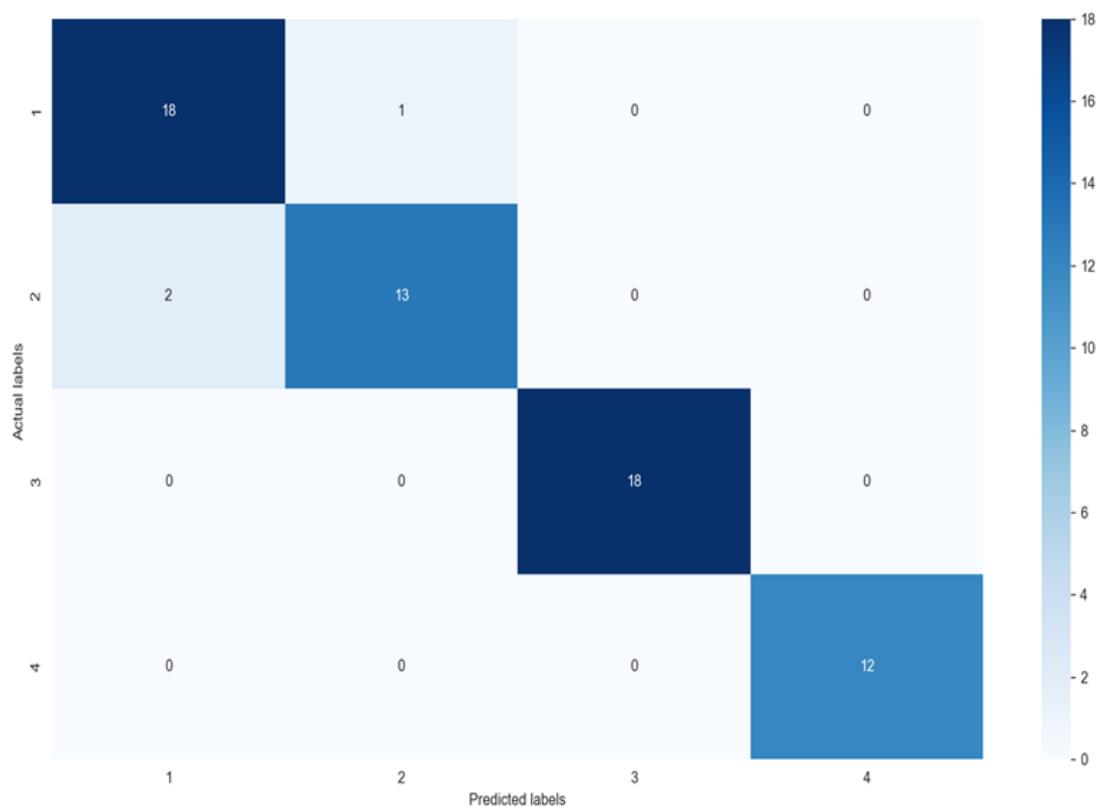
Hình 3.6: Biểu đồ số lượng các mẫu

Các thuật toán phân loại tiêu chuẩn khó có thể tối ưu được khi áp dụng trực tiếp cho dữ liệu chuỗi thời gian thô. Đầu tiên, tác giả chia dữ liệu thành các cửa sổ 3 giây. Sau đó, tác giả tạo các tính năng mới bằng cách tổng hợp 30 mẫu thô có trong mỗi phân đoạn 3 giây này. Để gán nhãn lớp cho các tính năng được chuyển đổi, tác giả sẽ lấy từ thẻ đa số trong cửa sổ đó. Ví dụ: tập dữ liệu thô có 100 hàng dữ liệu tuần tự. Vì vậy, sau khi tạo cửa sổ và tổng hợp (sử dụng kích thước cửa sổ = 30), nó sẽ được chuyển thành 10 hàng. Có một điều nữa, tác giả lấy các cửa sổ chồng chéo với 50% độ phủ. Điều này đảm bảo rằng mọi hàng tiếp theo trong tập dữ liệu được chuyển đổi cũng có một số thông tin từ dữ liệu trong cửa sổ trước đó. Sau khi trích xuất tính năng, tác giả sẽ sử dụng ma trận tương quan Hình ?? để đánh giá các tính năng.

Tác giả sẽ loại bỏ đi những tính năng có hệ số tương quan  $> 0.95$  để giảm thiểu thích thước dữ liệu. Kết quả có độ chính xác lên tới 96% trong việc đánh giá 4 tư thế ngủ từ tập kiểm tra chiếm 25% dữ liệu thu thập bằng cách áp dụng mô hình SVM. Đây mới chỉ là những kết quả bước đầu với số lượng mẫu còn hạn chế và chỉ tập trung vào 4 tư thế cơ bản. Nếu chỉ so sánh việc phát hiện 4 tư thế thì độ chính xác thì kết quả của 2 phương pháp tương đương nhau. Tuy nhiên, nhược điểm của phương pháp đánh giá ngưỡng là khó phát hiện những chuyển động đột ngột, bất thường trong lúc ngủ mà hiện tại tác giả coi đây là những tư thế chưa xác định. Để giải quyết vấn đề này thì học máy được coi là phương pháp hiệu quả nhất. Tư thế nằm ngửa và nghiêng trái đang có sự nhầm lẫn có thể do bước loại bỏ các dữ liệu hoặc do bước đánh nhãn cho 1 cửa sổ Hình ??.



Hình 3.7: Ma trận tương quan giữa các tính năng



Hình 3.8: Ma trận nhầm lẫn

## **Chương 4**

# **Kết luận**

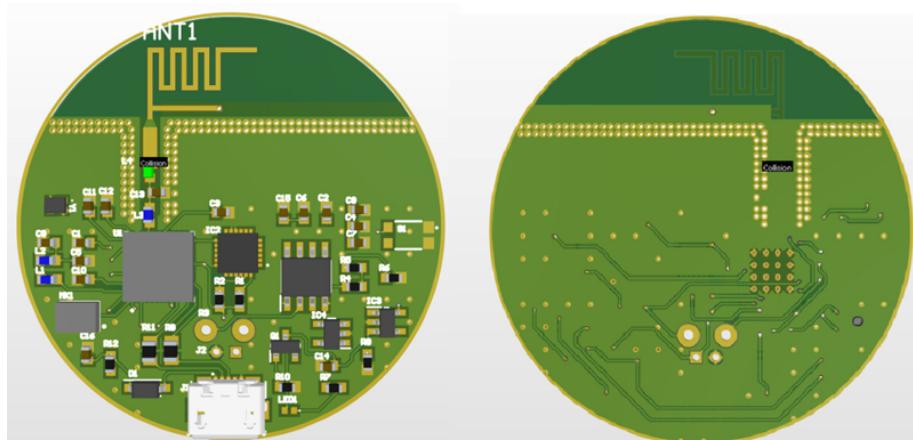
Sau khi hoàn thành khóa luận này, tác giả đã có thêm những kiến thức, kinh nghiệm như:

- Cách tìm, đọc, nghiên cứu tài liệu.
- Tổ chức, làm việc của 1 nhóm nghiên cứu.
- Hiểu biết thêm và thiết kế và lập trình mạch.
- Kinh nghiệm trong lập trình phần mềm.
- Kiến thức cơ bản về học máy.v. Kiến thức cơ bản về học máy.
- Ứng dụng vào công việc hiện tại.

Tuy nhiên đây mới chỉ là những bước đầu khai phá những kiến thức mới trong một định hướng nghiên cứu đòi hỏi sự hợp tác của nhiều ngành đặc biệt là ngành y học. Qua những điều đã đạt được, tác giả và nhóm nghiên cứu sẽ dần dần đúc rút thêm nhiều kinh nghiệm, kiến thức và kỹ năng để sớm hoàn thiện mục tiêu trong thời gian tới.

### **Hướng phát triển trong thời gian tới**

Đây là những tìm hiểu bước đầu của tác giả về thiết bị chẩn đoán chứng ngưng thở khi ngủ OSA vì vậy còn nhiều hạn chế về phần cứng và phần mềm. Hiện nay tác giả mới chỉ đánh giá các tư thế ngủ thông qua các ngưỡng của dữ liệu của 3 trực cảm biến. Điều này khiến việc trả về kết quả có thể chưa đạt độ chính xác cao. Vì vậy tác giả đã tìm hiểu các mô hình học máy để đánh giá 1 cách chính xác. Hơn nữa việc bật màn hình



Hình 4.1: Mạch thiết kế trên phần mềm Altium

điện thoại liên tục cũng làm hạn chế trải nghiệm của người dùng. Tác giả đã có những ý tưởng để cải thiện hiệu năng và dễ thao tác hơn đó với ứng dụng trên điện thoại. Đó là tích hợp thêm module wifi trên phần cứng từ đó tác giả đẩy dữ liệu trực tiếp lên phía server. Sau đó thiết bị trên điện thoại sẽ theo dõi trực tiếp từ server rồi hiển thị thông qua các ứng dụng chạy ngầm (foreground service android). Điều này có thể giúp người dùng tránh bật ứng dụng liên tục nhưng cũng khiến thời lượng pin của thiết bị giảm xuống. Như đã đề cập ở chương 2 tương lai tác giả sẽ có định hướng áp dụng các mô hình học máy cho đánh giá tư thế ngủ nữa là đánh giá chỉ số AHI dựa trên các thông số thu nhận từ các cảm biến. Tác giả cũng tìm hiểu nhiều cách để lấy được bộ dữ liệu và đánh nhãn thì chủ yếu các tác giả đánh nhãn bằng phương pháp thủ công như dùng máy ảnh để ghi lại, hoặc lấy nhãn của 1 hành động trong 1 thời gian cố định hoặc đánh trực tiếp trên thiết bị di động. Từ đó tác giả sẽ lên ý tưởng thiết kế tích hợp nút bấm trên ứng dụng điện thoại phục vụ việc lấy nhãn cho tập dữ liệu. Về phía phần mạch, nhóm tác giả đang phát triển và có những bước vẽ trên phần mềm Altium.

Trong thời gian tương lai, tác giả sẽ hoàn thành mạch in và thử nghiệm trên đó. Mục tiêu cuối cùng là ứng dụng được sản phẩm vào thực tiễn để có thêm cơ sở giúp các bác sĩ chẩn đoán sớm được những bệnh nhân mắc chứng OSA. Để hoàn thành được mục tiêu đó, trong tương lai, cả nhóm cần hoàn thiện các vấn đề còn tồn đọng và phát triển thêm những tính năng mới. Vấn đề áp dụng các mô hình học máy sẽ được nhóm tác giả tiếp tục phát triển không chỉ dừng lại ở đánh giá tư thế ngủ mà còn đánh giá chỉ số AHI.