

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

**TRẦN HỒNG HẢI**

**NGHIÊN CỨU XÂY DỰNG MẠNG CẢM BIẾN KHÔNG DÂY DỰA  
TRÊN GIAO THỨC LEACH VÀ ZIGBEE**

**LUẬN VĂN THẠC SĨ CÔNG NGHỆ THÔNG TIN**

**Hà Nội – Năm 2018**

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

**TRẦN HỒNG HẢI**

**NGHIÊN CỨU XÂY DỰNG MẠNG CẢM BIẾN KHÔNG DÂY DỰA  
TRÊN GIAO THỨC LEACH VÀ ZIGBEE**

**Ngành: Công nghệ thông tin**

**Chuyên ngành: Truyền DL & MMT**

**Mã số:**

**LUẬN VĂN THẠC SĨ CÔNG NGHỆ THÔNG TIN**

**NGƯỜI HƯỚNG DẪN KHOA HỌC: TS. Nguyễn Hoài Sơn**

**Hà Nội – Năm 2018**

## LỜI CAM ĐOAN

Tôi xin cam đoan đây là công trình nghiên cứu của riêng tôi. Các số liệu, kết quả nêu trong luận văn là trung thực và chưa được công bố trong các công trình khác. Nếu không đúng như đã nêu trên, tôi xin hoàn toàn chịu trách nhiệm về đề tài của mình.

Học viên

**Trần Hồng Hải**

## LỜI CẢM ƠN

Được trường Đại học Quốc Gia Hà Nội và Khoa Công Nghệ Thông Tin cho phép tôi nghiên cứu, viết luận văn về đề tài : “Nghiên cứu xây dựng mạng cảm biến không dây dựa trên giao thức LEACH và Zigbee”. Đầu tiên tôi xin chân thành cảm ơn quý thầy cô đã giảng dạy, chỉ dẫn tôi trong thời gian tôi học tập tại trường. Những kiến thức quý báu của thầy cô đã giúp tôi rất nhiều trong học tập cũng như trong cuộc sống. Tiếp theo tôi xin gửi lời cảm ơn sâu sắc nhất đến thầy Nguyễn Hoài Sơn. Cảm ơn thầy đã gắn bó, chỉ dạy tận tình tôi trong quá trình tôi nghiên cứu và viết luận văn. Một lần nữa xin cảm ơn quý thầy quý cô trong khoa và nhà trường. Luận văn thạc sỹ này được thực hiện dưới sự tài trợ từ đề tài NCKH cấp ĐHQGHN, mã số đề tài: QG.16.30.

Học viên

Trần Hồng Hải

# MỤC LỤC

MỞ ĐẦU .....	1
CHƯƠNG I: TỔNG QUAN VỀ MẠNG CẢM BIẾN .....	4
1.1. Giới thiệu về mạng cảm biến: .....	4
1.2. Cấu trúc của mạng cảm biến: .....	6
1.3. Kiến trúc giao thức của mạng cảm biến không dây .....	7
1.4. Ứng dụng của mạng cảm biến:.....	9
CHƯƠNG II: CÁC GIAO THỨC ĐỊNH TUYẾN MẠNG CẢM BIẾN.....	12
2.1 Tổng quan:.....	12
2.2 Giao thức LEACH:.....	12
2.3 Giao thức cải tiến LEACH-C:.....	16
2.4 ZigBee: .....	17
Chương III. XÂY DỰNG MẠNG CẢM BIẾN DỰA TRÊN LEACH VÀ ZIGBEE .....	21
3.1. Đặt vấn đề: .....	21
3.2. Giải pháp đề xuất: .....	22
3.3. Cách thức triển khai giải pháp: .....	23
Chương IV. XÂY DỰNG HỆ THỐNG VÀ ĐÁNH GIÁ GIẢI PHÁP .....	30
4.1. Tổng quan hệ thống:.....	30
4.2. Các thiết bị phần cứng:.....	30
4.2.1. Thiết bị truyền thông Xbee: .....	30
4.2.2. Bo mạch Arduino Nano: .....	32
4.2.3. Cài đặt hệ thống cơ bản.....	33
4.3. Xây dựng nút SINK: .....	39
4.4. Xây dựng nút Cluster Head:.....	41
4.5. Lắp đặt chạy thử hệ thống:.....	41
4.6. Lắp đặt hệ thống đo mức tiêu thụ điện: .....	41
Chương V. KẾT LUẬN .....	53
TÀI LIỆU THAM KHẢO .....	55

## DANH MỤC HÌNH ẢNH

Hình 1.1 : Cấu trúc mạng cảm biến không dây ( <a href="https://www.researchgate.net">https://www.researchgate.net</a> ) ..	6
Hình 1.2 Sensor node .....	7
Hình 1.3 Kiến trúc giao thức của mạng cảm biến không dây .....	7
Hình 1.4 SmartHouse( <a href="http://www.vietnamnet.vn">www.vietnamnet.vn</a> ) .....	9
Hình 2.1. Phân loại và so sánh các giao thức chọn đường trong WSN [6] .....	12
Hình 2.2 : Mô hình giao thức LEACH ( <a href="http://www.spiroprojects.com">http://www.spiroprojects.com</a> ) .....	13
Hình 2.3 : Ngưỡng thiết lập Cluster Head .....	15
Hình 2.4 : Lưu đồ thuật toán cài đặt LEACH( <a href="http://www.slideshare.net">www.slideshare.net</a> ) .....	15
Hình 2.5 : Lưu đồ thuật toán giai đoạn ổn định LEACH( <a href="http://www.slideshare.net">www.slideshare.net</a> ) .	16
Hình 2.6 : Giai đoạn cài đặt của LEACH-C ( <a href="http://www.slideshare.net">www.slideshare.net</a> ) .....	17
Hình 2.7 : Cấu trúc của mạng ZigBee( <a href="http://adlt.com.au">adlt.com.au</a> ) .....	18
Hình 2.8 : Mô hình mạng ZigBee ( <a href="http://arduino.vn/">http://arduino.vn/</a> ) .....	19
Hình 3.1. Sơ đồ chức năng của nút chủ SINK .....	25
Hình 3.2. Khung dữ liệu nút SINK nhận được .....	25
Hình 3.3. Khung dữ liệu nút SINK gửi đi .....	26
Hình 3.4. Sơ đồ chức năng nút thành phần .....	27
Hình 3.5. Khung dữ liệu Broad Cast của Cluster Head .....	28
Hình 3.6. Khung dữ liệu được gửi từ nút thành phần đến Cluster Head .....	28
Hình 4.1 Sơ đồ hệ thống .....	30
Hình 4.2 : Thiết bị Xbee ( <a href="http://arduino.vn/">http://arduino.vn/</a> ) .....	31
Hình 4.3 : Bộ kết hợp của Xbee và Arduino ( <a href="http://arduino.vn/">http://arduino.vn/</a> ) .....	31
Hình 4.4 : Cấu trúc Xbee ( <a href="http://arduino.vn/">http://arduino.vn/</a> ) .....	32
Hình 4.5 : Arduino Nano ( <a href="http://arduino.vn/">http://arduino.vn/</a> ) .....	32
Hình 4.6: Sơ đồ cấu trúc Arduino Nano ( <a href="http://arduino.vn/">http://arduino.vn/</a> ) .....	33
Hình 4.7 : Để chuyển đổi từ 2.0 sang 2.5 ( <a href="http://arduino.vn/">http://arduino.vn/</a> ) .....	33
Hình 4.8 : Xbee và Arduino Nano trên Bread Board .....	34
Hình 4.9 : Bộ cấu hình cho Xbee .....	34
Hình 4.10 : Nhận diện Xbee .....	35
Hình 4.11 : Cấu hình Xbee .....	36
Hình 4.12 : Sơ đồ mạch kết nối Xbee và Arduino Nano ( <a href="http://arduino.vn/">http://arduino.vn/</a> ) ....	37

Hình 4.13 : Code cho nút SINK ( Coordinator ).....	38
Hình 4.15 : Giao diện Arduino IDE.....	40
Hình 4.19: Pin sạc dự phòng.....	42
Hình 4.19. Gói tin trong ZigBee thuần.....	42
Hình 4.20. Gói tin trong phương pháp mới của tôi.....	43
Hình 4.21. Mô hình 7 node ZigBee thuần.....	43
Hình 4.22. Mô hình mạng của phương pháp mới với 7 node.....	44

## **CÔNG THỨC TRONG LUẬN VĂN**

Công thức (1) : Xác định Cluster Head trong giao thức LEACH ... Trang 15

Công thức (2) : Tính hiệu suất truyền dữ liệu thành công ..... Trang 29



## MỞ ĐẦU

### 1. Lý do chọn đề tài:

Hiện nay khoa học kỹ thuật được đưa vào áp dụng cho công nông nghiệp rất nhiều. Với nhu cầu tiêu thụ thực phẩm sạch an toàn ngày càng cao cho nên những vườn trồng hay trang trại có quy mô được mở ra. Nhằm giúp cho người trồng quản lý được môi trường của vườn trồng nhanh và chính xác thì những thiết bị cảm biến được sử dụng. Tính cấp thiết đối với thực tế, vì vậy tôi đã tìm hiểu và nghiên cứu về đề tài xây dựng mạng cảm biến với mong muốn mang lại hiệu quả hơn cho người sử dụng.

Mạng cảm biến không dây là hệ thống tập hợp, liên kết nhiều cảm biến với nhau sử dụng các liên kết không dây để phối hợp thực hiện nhiệm vụ thu thập thông tin dữ liệu với quy mô lớn trong bất kỳ điều kiện và ở bất kỳ vùng địa lý nào. Mạng cảm biến không dây có thể liên kết trực tiếp với nút quản lý giám sát trực tiếp hay gián tiếp thông qua một điểm thu phát và môi trường mạng công cộng như Internet hay vệ tinh. Mỗi một cảm biến chịu một hoặc nhiều nhiệm vụ khác nhau tùy thuộc vào ứng dụng. Hiện nay mạng cảm biến không dây được áp dụng trong nhiều lĩnh vực ví dụ như: y tế, công nghiệp, nông nghiệp, nghiên cứu,... Các thiết bị cảm biến không dây liên kết thành một mạng đã tạo ra nhiều khả năng mới cho con người. Các đầu đo với bộ vi xử lý rất nhỏ gọn tạo nên một thiết bị cảm biến không dây có kích thước rất nhỏ, tiết kiệm về không gian. Chúng có thể hoạt động trong môi trường dày đặc với khả năng xử lý tốc độ cao. Ngày nay, các mạng cảm biến không dây được ứng dụng trong nhiều lĩnh vực như nghiên cứu vi sinh vật biển, giám sát việc chuyên chở các chất gây ô nhiễm, kiểm tra giám sát hệ sinh thái và môi trường sinh vật phức tạp, điều khiển giám sát trong công nghiệp và trong lĩnh vực quân sự, an ninh quốc phòng hay các ứng dụng trong đời sống hàng ngày.

Công nghệ kỹ thuật phát triển mạnh như hiện nay mạng cảm biến không dây được cải tiến liên tục mang lại hiệu quả to lớn cho nhiều ngành nghề trong

nhều lĩnh vực. Đi kèm với công nghệ 4.0 và công nghệ IOT đang phát triển vượt bậc cho ta thấy tương lai tươi sáng của mạng cảm biến không dây.

Xbee là một thiết bị truyền thông không dây được sử dụng rộng rãi trong mạng cảm biến không dây. Xbee có những thế mạnh cần thiết như: sự tin cậy, tính mở rộng, dễ sử dụng, tiết kiệm năng lượng,... Xbee sử dụng giao thức Zigbee cơ bản để hình thành mạng cảm biến không dây. Với những thế mạnh của Xbee nêu ở trên đề tài của tôi lựa chọn Xbee là thiết bị để xây dựng mạng cảm biến không dây ZigBee.

Đề tài này có thể giúp tôi tìm hiểu và vận dụng các kiến thức vào thực tế, cùng với đó tích lũy thêm kinh nghiệm và chuyên môn. [2]

## **2. Mục tiêu nghiên cứu:**

Hiện nay nhiều những thiết bị cảm biến và nhiều mạng cảm biến được thiết kế ra. Mỗi mạng cảm biến lại có thế mạnh riêng, cách thức sử dụng riêng. Vì vậy trong đề tài tôi chọn hai giao thức mạng cảm biến dùng nhiều nhất và phù hợp với điều kiện ở Việt Nam là LEACH và Zigbee. Mục tiêu của tôi là đưa ra phương pháp để kết hợp thế mạnh của hai giao thức mạng cảm biến lại với nhau nhằm xây dựng một mạng cảm biến hoạt động hiệu quả hơn.

Đề tài tập trung nghiên cứu và thay đổi phương thức định tuyến của mô hình mạng cảm biến ZigBee theo phương thức định tuyến của giao thức LEACH. Hệ thống được áp dụng trên 7 thiết bị truyền phát dữ liệu Xbee, sử dụng Arduino để định tuyến đường truyền gói tin. Phân tích hiệu quả của mô hình mạng lại về ba mặt : hiệu quả truyền tin, kéo dài sự sống của mạng, hiệu suất tiêu thụ năng lượng.

Kiến thức về mạng cảm biến không dây tôi có được học trong chương trình đào tạo cao học của nhà trường. Tài liệu nghiên cứu chủ yếu được lấy từ những bài báo trong và ngoài nước và những kiến thức thu thập được từ các trang chia sẻ thông tin trên mạng Internet.

Đi sâu tìm hiểu về cách thức đóng gói, định tuyến gói tin của mạng cảm biến không dây nhằm thay đổi theo cách thức mình mong muốn.

### 3. Cấu trúc đề tài:

Qua thời gian tìm hiểu và nghiên cứu tôi đã viết luận văn với các nội dung như sau:

- Chương một là tổng quan về mạng cảm biến : Chương này tập trung giới thiệu về mạng cảm biến không dây, một số mạng cảm biến được áp dụng hiện nay.
- Chương hai là mạng cảm biến liên quan: Giới thiệu về mạng cảm biến ZigBee và hai giao thức liên quan trong đề tài là LEACH, LEACH-C.
- Chương ba là đặt vấn đề, đưa ra giải pháp đề xuất và phương pháp thực hiện giải pháp trên.
- Chương bốn là xây dựng mạng cảm biến không dây dựa trên giao thức LEACH và ZigBee, đánh giá hiệu quả mang lại.
- Chương năm là kết luận : Nêu ra những thế mạnh và kết quả của đề tài, rút ra những kinh nghiệm và những khó khăn, những vấn đề chưa giải quyết được. Phương hướng tiếp theo sẽ như thế nào.

# CHƯƠNG I: TỔNG QUAN VỀ MẠNG CẢM BIẾN

## 1.1. Giới thiệu về mạng cảm biến:

Mạng cảm biến không dây (Wireless Sensor Network) là tập hợp các thiết bị cảm biến sử dụng các liên kết không dây để phối hợp thực hiện nhiệm vụ thu thập thông tin dữ liệu với quy mô lớn trong bất kỳ điều kiện và ở bất kỳ vùng địa lý nào. Mạng cảm biến không dây có thể liên kết trực tiếp với nút quản lý giám sát trực tiếp hay gián tiếp thông qua một điểm thu phát (Sink) và môi trường mạng công cộng như Internet hay vệ tinh. Các nút cảm biến không dây có thể được triển khai cho các mục đích chuyên dụng như điều khiển giám sát và an ninh; kiểm tra môi trường; thu thập thông tin môi trường; giám sát sức khỏe cho bệnh nhân trong y tế; ... Thế mạnh chủ yếu của chúng sự linh hoạt cũng như không dây nên có thể triển khai ở những khu vực đặc biệt mà thiết bị có dây không thể xây dựng.

Các thiết bị cảm biến không dây được liên kết với nhau tạo thành một mạng hoặc các cụm mạng có cấu trúc theo chuẩn nhất định. Giúp cho chúng ta có thể tùy chỉnh về mô hình, cách thức hoạt động, quy mô của mạng. Các thiết bị thường được thiết kế nhỏ gọn, tiết kiệm về không gian và năng lượng tiêu thụ thấp. Hiện nay khoa học tiên tiến đã đẩy khả năng xử lý của chip điện tử lên cao vì vậy các thiết bị cảm biến tuy nhỏ gọn nhưng có thể thực hiện được rất nhiều công việc. Với độ bền thiết bị được đảm bảo cho nên mạng cảm biến không dây thường hoạt động liên tục, đưa lại cho chúng ta những thông tin cần thiết và kịp thời. Hệ thống mạng cảm biến được áp dụng rất nhiều trong các lĩnh vực khoa học cũng như cuộc sống. [2] [7]

Mạng cảm biến không dây có nhưng đặc điểm nổi bật như sau: [2] [6]

### - **Kích thước vật lý nhỏ gọn:**

Các thiết bị trong mạng cảm biến không dây thường được thiết kế nhỏ gọn để tiện dụng. Mặc dù kích thước luôn tỉ lệ nghịch với tốc độ xử lý, giới hạn lưu trữ, thời gian hoạt động. Nhưng với những mục đích mình mong muốn thì việc phát triển và thiết kế sao cho hợp lý và hài hòa nhất. Với công nghệ tiên

tiền bây giờ các thiết bị trong mạng cảm biến không dây cũng khá đủ cho nhu cầu của người dùng: kích thước bé, năng lượng tiêu thụ thấp, xử lý nhanh,...

- ***Hoạt động đồng thời với độ tập trung cao***

Do nhu cầu của người dùng nên việc một mạng cảm biến không dây thường hoạt động liên tục và chính xác cao trong thời gian dài. Ví dụ như việc giám sát sức khỏe người bệnh trong y tế chẳng hạn, người dùng phải được theo dõi thường xuyên và liên tục. Những thông số môi trường hay thông tin dữ liệu thường được người dùng yêu cầu tức thì và chính xác cho nên các thiết bị mạng cảm biến không dây được thiết kế với những tiêu chí như trên. Việc thiết kế phần cứng ảnh hưởng trực tiếp trong quá trình trên nhưng khi chúng ta thiết kế phần mềm hợp lý cũng mang lại hiệu quả khá bất ngờ.

- ***Tính đa dạng trong thiết kế và sử dụng***

Hiện nay có rất nhiều dạng cảm biến, những cảm biến được dành riêng cho một số lĩnh vực nhất định. Những hệ thống được thiết kế cho những công việc đặc thù như đo lượng dầu trong nước hay đo nồng độ khí CO<sub>2</sub> trong không khí chẳng hạn. Cho nên những người phát triển mạng cảm biến không dây đã đưa ra những giải pháp thích hợp là xây dựng một nền tảng chung có thể áp dụng, kết hợp nhiều thiết bị lại với nhau. Sử dụng linh hoạt phần mềm để có thể thích ứng với phần cứng riêng nhằm đưa lại hiệu quả công việc.

- ***Hoạt động tin cậy***

Những thông tin dữ liệu mà mạng cảm biến mang lại rất quan trọng và cần thiết. Đối với những lĩnh vực đặc biệt thì điều đó là tối cần thiết. Hiện nay khoa học công nghệ phát triển đã thiết kế ra những thiết bị có độ tin cậy cao và cung cấp thông tin nhanh chóng. Như việc linh động đã nêu ở trên thì việc người dùng sử dụng phần mềm thích hợp sẽ cho chúng ta hiệu quả cao, độ tin cậy thông tin cao. Có rất nhiều yếu tố mang lại sự tin cậy cho mạng cảm biến không dây cũng như có rất nhiều yếu tố bên ngoài gây ra việc giảm độ tin cậy. Trong mỗi trường hợp chúng ta phải đưa ra những giải pháp hợp lý để sự tin cậy của mạng cảm biến không dây là tốt nhất.

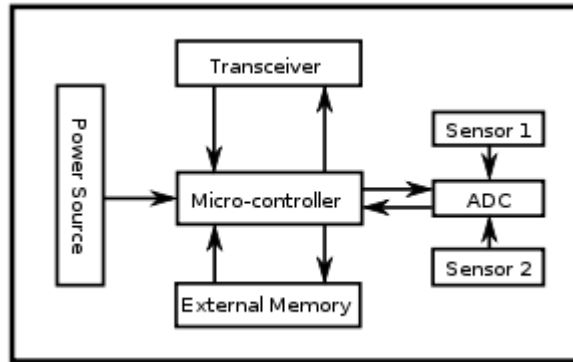
## 1.2. Cấu trúc của mạng cảm biến:

Một mạng cảm biến không dây thường có số lượng lớn các nút được phủ trong một vùng, khu vực nào đấy mà chúng ta muốn thăm dò hay thu thập thông tin dữ liệu. Do có sự gắn kết trong mạng nên các nút có thể tùy chỉnh vị trí sao cho phù hợp với địa hình cũng như yêu cầu người dùng. Với khả năng mở rộng cho nên mạng cảm biến không dây có thể tiếp cận những khu vực nguy hiểm hoặc địa hình đặc thù. Khả năng liên kết cũng như công tác làm việc của các cảm biến không dây chính là đặc trưng cơ bản nhất. Với số lượng lớn các cảm biến không dây được triển khai gần nhau thì truyền thông đa liên kết được lựa chọn để công suất tiêu thụ là nhỏ nhất (so với truyền thông đơn liên kết) và mang lại hiệu quả truyền tin hiệu tốt hơn so với truyền khoảng cách xa.



Hình 1.1 : Cấu trúc mạng cảm biến không dây (<https://www.researchgate.net>)

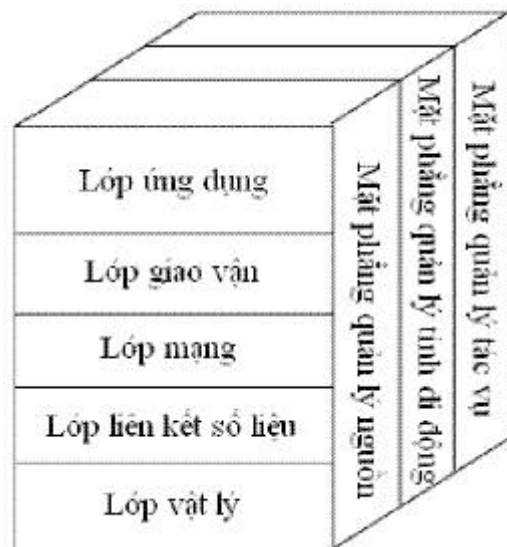
Cấu trúc cơ bản của mạng cảm biến không dây được thể hiện trên hình 1.1. Các nút cảm biến được xây dựng trong một trường cảm biến chính là khu vực được bao viền xanh trên hình. Mỗi nút cảm biến được sắp đặt trong mạng có khả năng thu thập thông tin dữ liệu, định tuyến gói tin về nút chủ là bộ thu nhận (Sink) để chuyển tới người dùng (User) qua internet hoặc một môi trường khác. Nút Sink có thể yêu cầu việc định tuyến để các nút cảm biến truyền bản tin theo mong muốn. Số liệu được nút Sink nhận về thường không có một định dạng cụ thể, có thể tùy chỉnh sao cho phù hợp với mục đích của mình.



Hình 1.2 Sensor node

Mỗi nút cảm biến bao gồm bốn thành phần cơ bản là: hệ thống cảm biến, bộ xử lý, bộ phận lưu trữ nếu cần, thiết bị thu phát không dây và nguồn năng lượng. Tùy theo nhu cầu người dùng nên có những nút cảm biến còn được gắn thêm các thiết bị như hệ thống định vị, hệ thống năng lượng mặt trời,... Các thành phần cơ bản trong một nút cảm biến được thể hiện trên hình 1.2. Hệ thống cảm biến bao gồm các đầu đo Sensor và một bộ chuyển đổi tương tự/số ADC. Các tín hiệu tương tự được thu nhận từ đầu đo được chuyển sang tín hiệu số bằng bộ chuyển đổi ADC, những tín hiệu số được đưa vào bộ xử lý. Bộ xử lý thường được gắn thêm một bộ nhớ nhỏ, nhằm xử lý tín hiệu số và truyền dữ liệu cho thiết bị thu phát không dây. Ở đây bộ xử lý có thể định dạng gói tin cũng như định tuyến gói tin cho thiết bị không dây. Thiết bị không dây có nhiệm vụ thu phát các thông tin dữ liệu giữa các nút cảm biến với nhau. [2] [6]

### 1.3. Kiến trúc giao thức của mạng cảm biến không dây



Hình 1.3 Kiến trúc giao thức của mạng cảm biến không dây

Kiến trúc giao thức được sử dụng trong nút chủ (Sink) và tất cả các nút cảm biến được thể hiện trên hình 1.3. Kiến trúc giao thức bao gồm lớp ứng dụng (Application Layer), lớp giao vận (Transport Layer), lớp mạng (Network Layer), lớp liên kết số liệu (Datalink Layer), lớp vật lý (Physical Layer), mặt phẳng quản lý năng lượng (Power Management Plane), mặt phẳng quản lý di động (Mobility Management Plane) và mặt phẳng quản lý nhiệm vụ (Task Management Plane). Kiến trúc gồm ba mặt phẳng quản lý là : Mặt phẳng quản lý nguồn, mặt phẳng quản lý tính di động, mặt phẳng quản lý tác vụ.

- *Mặt phẳng quản lý nguồn* : Quản lý cách cảm biến sử dụng nguồn năng lượng của nó. Ví dụ : một nút cảm biến có thể bật hoặc tắt bộ thu nhận sau khi nhận được một bản tin. Các nút cảm biến có thể broadcast đến các nút bên cạnh rằng mình không đủ năng lượng để thực hiện công việc được nữa.
- *Mặt phẳng quản lý tính di động* : Có nhiệm vụ quản lý tính di động của nút. Có thể phát hiện các nút bên cạnh và thông báo vị trí của nó cho các nút đây.
- *Mặt phẳng quản lý tác vụ* : Làm nhiệm vụ sắp xếp các công việc giữa các nút trong vùng liên quan.
- *Lớp vật lý*: Có nhiệm vụ lựa chọn tần số, tạo ra tần số sóng mang, phát hiện tín hiệu, điều chế và mã hóa tín hiệu. Trong mạng cảm biến thì băng tần sử dụng nhiều nhất là ISM 915 MHZ.
- *Lớp liên kết dữ liệu*: Lớp này có nhiệm vụ liên kết dữ liệu, phát hiện dữ liệu hay các frame dữ liệu cần thiết. Vì một mạng cảm biến thường nhận được nhiều tín hiệu tạp âm nên nhiệm vụ của lớp này rất cần thiết.
- *Lớp mạng*: Lớp mạng được thiết kế dùng để phân biệt các mạng cảm biến khác nhau. Trong lớp này có những số liệu mà người dùng cài đặt để xây dựng một mạng cảm biến riêng cho mình.
- *Lớp truyền tải*: Chỉ cần thiết khi hệ thống được thiết kế cho việc truy cập qua internet hoặc mạng bên ngoài.



- *Lớp ứng dụng*: Tùy theo nhiệm vụ của cảm biến, các loại ứng dụng, phần mềm khác nhau được xây dựng và sử dụng ở lớp ứng dụng này. [2]

#### 1.4. Ứng dụng của mạng cảm biến:

- **Giám sát và điều khiển công nghiệp**

Hiện nay nền công nghiệp cực kỳ phát triển, rất nhiều lĩnh vực cần đến tự động hóa. Hay có những công việc phải điều khiển từ xa, sức khỏe con người không thể lại gần. Những khu vực nguy hiểm cần phải giám sát thường xuyên như hầm mỏ, khu sản xuất vật liệu nguy hiểm, khu vực thử nghiệm thuốc súng thuốc nổ. Mạng cảm biến không dây áp dụng rất nhiều trong công nghiệp, đem lại nguồn thu lớn lao cho con người, giảm thiểu khả năng tai nạn ảnh hưởng sức khỏe người lao động. Một ví dụ đơn giản và dễ thấy nhất là hệ thống phòng cháy chữa cháy trong công ty bạn chẳng hạn. Đó cũng là một cảm biến nhiệt độ và cảm biến khói trong phòng. Khi phát hiện ra nó sẽ tự động phun nước dập lửa.

- **Tự động hoá gia đình và điện dân dụng**



Hình 1.4 SmartHouse([www.vietnamnet.vn](http://www.vietnamnet.vn))

Hiện nay SmartHouse chắc không xa lạ đối với chúng ta. Hệ thống nhà thông minh bao gồm một hoặc nhiều mạng cảm biến khác nhau. Phục vụ rất nhiều công việc cho con người. Một ứng dụng được điều khiển chung từ xa,

máy nghe nhạc, dàn âm thanh nổi và các thiết bị điện tử gia đình khác hay các bóng đèn, các cánh cửa, và các ổ khoá cũng được trang bị với kết nối mạng cảm biến không dây. Với hệ thống điều khiển từ xa bạn có thể điều khiển, quản lý ngôi nhà của bạn. Tuy nhiên, khả năng hấp dẫn nhất là sự kết hợp thông minh từ nhiều dịch vụ, giống như các cánh cửa tự động đóng khi TV được bật, hoặc có thể tự động ngưng hệ thống giải trí gia đình khi có điện thoại hoặc chuông cửa reo. Mục đích lớn của các mạng cảm biến không dây trong gia đình được mong chờ là mức tiêu thụ điện thấp và tính ổn định cao là điều kiện thiết yếu của các mạng cảm biến không dây.

- **Triển vọng của mạng cảm biến không dây trong quân sự**

Các mạng cảm biến không dây là một phần cực kỳ quan trọng trong các ứng dụng quân sự ngày nay với các hệ thống ra lệnh, điều khiển, thu thập tin tức tình báo truyền thông, tính toán, theo dõi kẻ tình nghi, trinh sát và tìm mục tiêu. Các đặc tính triển khai nhanh chóng, tự tổ chức và khả năng chịu đựng lỗi của các mạng cảm biến cho thấy đây là một công nghệ đầy triển vọng trong lĩnh vực quân sự. Với đặc điểm nhỏ gọn thì một hệ thống theo dõi mục tiêu rất đơn giản mà khó có thể phát hiện được. Một số ứng dụng của mạng cảm biến là: kiểm tra lực lượng, trang bị, đạn dược, giám sát chiến trường, trinh sát vùng địa lý và lực lượng địch, tìm mục tiêu, đánh giá thiệt hại trận đánh, trinh sát và phát hiện các vũ khí hóa học - sinh học - hạt nhân.

- **Mạng cảm biến không dây trong y tế và giám sát sức khỏe**

Trong y tế hiện nay cũng áp dụng rất nhiều về mạng cảm biến không dây. Ví dụ như một hệ thống theo dõi sức khỏe bệnh nhân như: Nhịp tim, nồng độ cồn, nồng độ chất trong máu, hoạt động chân tay,... Vì những đặc điểm cơ bản của mạng cảm biến không dây được nêu ra ở trên cho nên trong y tế rất được coi trọng và ứng dụng.

- **Mạng cảm biến không dây với môi trường và ngành nông nghiệp**

Trong nông nghiệp hiện nay đã và đang sử dụng những mạng cảm biến không dây lớn. Việc theo dõi thông số cho cây trồng luôn được ưu tiên hàng đầu. Ví dụ như việc theo dõi nhiệt độ, hoặc lưu lượng nước, nồng độ chất trong

dung dịch nuôi cây chẳng hạn. Với đặc điểm mở rộng và dễ lắp đặt thì việc áp dụng mạng không dây trên một diện tích trồng cây lớn là hoàn toàn hợp lý. Một số các ứng dụng về môi trường của mạng cảm biến không dây bao gồm theo dõi sự di chuyển của các loài chim, loài thú nhỏ, côn trùng; kiểm tra các điều kiện môi trường ảnh hưởng tới mùa màng và vật nuôi; tình trạng nước tưới; các công cụ vĩ mô cho việc giám sát mặt đất ở phạm vi rộng và thám hiểm các hành tinh; phát hiện hóa học, sinh học; tính toán trong nông nghiệp; kiểm tra môi trường không khí, đất trồng, biển; phát hiện cháy rừng; nghiên cứu khí tượng và địa lý; phát hiện lũ lụt; vẽ bản đồ sinh học phức tạp của môi trường và nghiên cứu ô nhiễm môi trường. [2]

## CHƯƠNG II: CÁC GIAO THỨC ĐỊNH TUYẾN MẠNG CẢM BIẾN

### 2.1 Tổng quan:

Mạng cảm biến không dây có nhiều điểm giống với mạng adhoc nhưng cũng có rất nhiều đặc tính riêng nên có thể phân loại thành một dạng mạng riêng. Với những đặc tính đã trình bày ở chương trước giúp chúng ta có thể thiết kế ra nhiều giao thức định tuyến cho mạng cảm biến không dây. Để thiết kế được giao thức định tuyến cho mạng cảm biến không dây cần phải nắm rõ những vấn đề liên quan đến mạng cảm biến. Trong mạng cảm biến không dây có ba giao thức định tuyến chính hay được dùng trong mạng cảm biến, đó là định tuyến ngang hàng hay gọi là trung tâm dữ liệu (data-centric-protocol), định tuyến phân cấp (hierarchical-protocol) và định tuyến dựa vào vị trí (location-based-protocol). [6]

Giao thức chọn đường	Giao thức trung tâm dữ liệu	Giao thức phân cấp	Giao thức dựa trên vị trí	Phân loại	Di chuyển	Dựa vào hỏi đáp	Kết hợp số liệu	Xác định vị trí	QoS	Độ phức tạp của trạng thái	Khả năng định cỡ	Đa đường	Dựa vào yêu cầu
SPIN	X			Ngang hàng	Có thể	Có	Có	Không	Không	Thấp	Hạn chế	Có	Có
Directed	X			Ngang hàng	Hạn chế	Có	Có	Có	Không	Thấp	Hạn chế	Có	Có
Diffusion Rumor	X			Ngang hàng	Rất hạn chế	Không	Có	Không	Không	Thấp	Tốt	Không	Có
GBR	X			Ngang hàng	Hạn chế	Không	Có	Không	Không	Thấp	Tốt	Không	Có
CADR	X			Ngang hàng	Không	Không	Có	Không	Không	Thấp	Hạn chế	Không	Không
COUGAR	X			Ngang hàng	Không	Không	Có	Không	Không	Thấp	Hạn chế	Không	Không

Hình 2.1. Phân loại và so sánh các giao thức chọn đường trong WSN [6]

### 2.2 Giao thức LEACH:

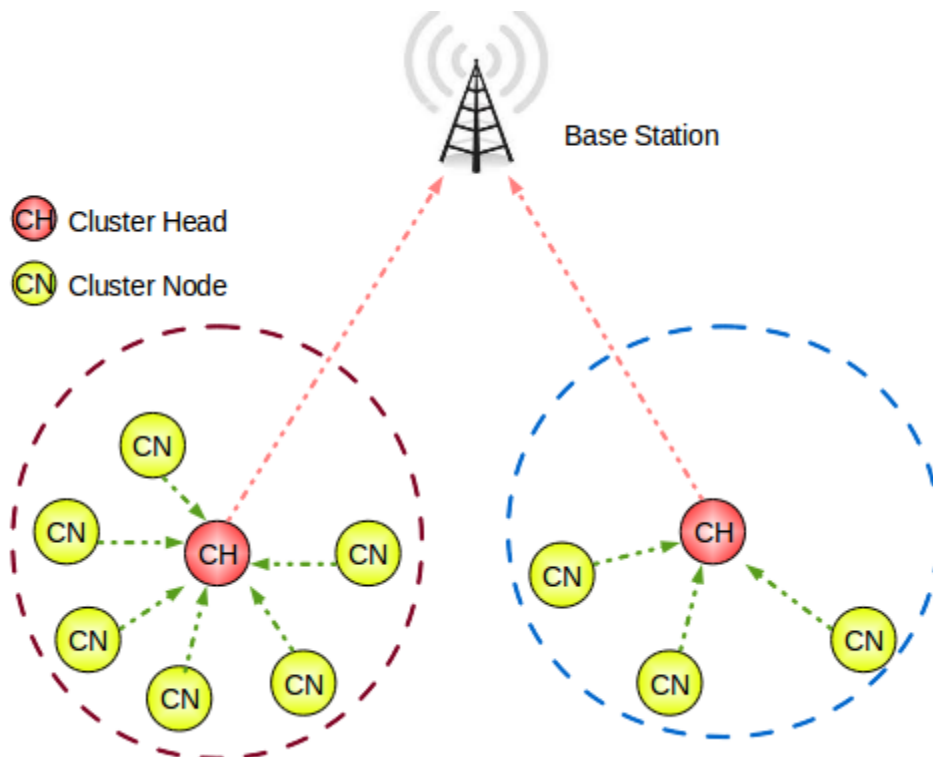
Những tiến bộ gần đây trong thông tin vô tuyến và điện tử đã cho phép phát triển các mạng cảm biến giá thành thấp. Mạng cảm biến có thể được sử dụng trong các ứng dụng khác nhau như theo dõi môi trường, y tế, trong quân sự hoặc sử dụng trong gia đình. Mạng cảm biến vô tuyến (WSN) bao gồm các nút

nhỏ có khả năng cảm biến, tính toán và trao đổi thông tin vô tuyến. Một số giao thức chọn đường, quản lý công suất và trao đổi số liệu đã được thiết kế cho mạng cảm biến với yêu cầu quan trọng nhất là tiết kiệm được năng lượng. Các giao thức chọn đường trong mạng cảm biến có thể khác nhau tùy theo ứng dụng và cấu trúc mạng.

LEACH là một giao thức được xây dựng theo loại phân cấp. LEACH được viết tắt từ “Low Energy Adaptive Clustering Hierarchy” nghĩa là “Phân cấp nhóm thích ứng công suất thấp”. Người phát minh ra LEACH là ông Heinzelman. Mục đích của LEACH là lựa chọn ngẫu nhiên các nút cảm biến làm các nút chủ cụm, do đó việc tiêu hao năng lượng khi liên lạc với nút gốc được trải đều cho tất cả các nút cảm biến trong mạng. Quá trình hoạt động của LEACH được chia thành hai bước là bước cài đặt và bước ổn định. Thời gian của bước ổn định kéo dài hơn so với thời gian của bước cài đặt. [4] [5]

Leach là một cụm dựa trên giao thức bao gồm các tính năng sau đây:

- + Ngẫu nhiên và tự cấu hình thành cụm.
- + Kiểm soát lượng dữ liệu truyền nhận.
- + Sử dụng phương tiện truyền thông năng lượng thấp.



Hình 2.2 : Mô hình giao thức LEACH (<http://www.spiroprojects.com>)

Trong giao thức LEACH có 3 thành phần đó là nút Sink đảm nhận vai trò là nút chủ thu thập thông tin của cả hệ thống mạng. Tiếp theo là nút Cluster Head chịu trách nhiệm là nút tập trung dữ liệu của một cụm các nút xung quanh sau đó chuyển tiếp lên nút Sink. Cuối cùng là nút Non Cluster Head chính là nút lá trong mạng cảm biến đã giới thiệu ở trên. Nút Non Cluster Head hoạt động chủ yếu là thu thập thông tin hoặc điều khiển một thiết bị nào đó được người dùng chỉ định. Hoạt động của LEACH được phân tách thành hai bước, bước và bước ổn định trạng thái. Ở trong bước cài đặt, các nhóm được tự tổ chức và tự lựa chọn các nút chính. Còn ở giai đoạn ổn định trạng thái, việc truyền số liệu thực sự về các trạm gốc được tiến hành. Bước cài đặt thường được triển khai khi mạng bắt đầu hoạt động và thời gian cài đặt thường rất ngắn để dành thời gian, năng lượng cho công việc chính của mạng cảm biến. [4] [5]

Vì sự tương quan dữ liệu giữa các nút gần nhau nên việc phân cụm trong mạng chính là cơ sở của LEACH. Điều này cho phép tất cả các dữ liệu từ các nút trong cụm có thể được xử lý tại nút cluster-head, giảm bớt những dữ liệu không cần thiết chuyển đến người dùng cuối. Vì vậy, các nút thành phần Non Cluster Head sẽ tiết kiệm năng lượng. Trong LEACH, các nút sẽ tự xét xem có trở thành Cluster Head hay không, các nút không trở thành Cluster Head sẽ nhận một nút Cluster Head nào đó để hình thành cụm. Tất cả các nút Non Cluster Head gửi dữ liệu của nó cho nút Cluster Head, trong khi nút Cluster Head phải nhận được dữ liệu từ tất cả các thành viên trong nhóm và truyền tải dữ liệu đến nút chủ ở xa. Cho nên nút Cluster Head sẽ tiêu tốn nhiều năng lượng hơn nút Non Cluster Head. Trong thực tế các nút cảm biến không dây thường có năng lượng có hạn, khi nút Cluster Head hết năng lượng sẽ dẫn tới việc cụm đó bị đình trệ và không hoạt động nữa. Cho nên LEACH đưa ra phương thức xoay vòng để các nút trong mạng đảm nhận công việc của Cluster Head trong một thời gian. Như vậy năng lượng tiêu thụ của việc vận chuyển dữ liệu lên nút chủ (SINK) và xử lý dữ liệu sẽ được chia đều cho tất cả các nút trong mạng. Điều này dẫn đến thời gian hoạt động và tính ổn định của mạng được kéo dài đến khi tất cả các nút hết năng lượng. [4] [5]

Đầu tiên các nút thành phần sẽ quyết định có trở thành Cluster Head hay không. Quyết định này dựa trên tỷ lệ phần trăm trở thành Cluster Head cho các nút do nút chủ quyết định và số lần trở thành Cluster Head. Quyết định này được thực hiện bởi n nút lựa chọn ngẫu nhiên một số từ 0 đến 1. Nếu con số sẽ thấp hơn một ngưỡng T(n), các nút sẽ trở thành Cluster Head. Ngưỡng được thiết lập như:

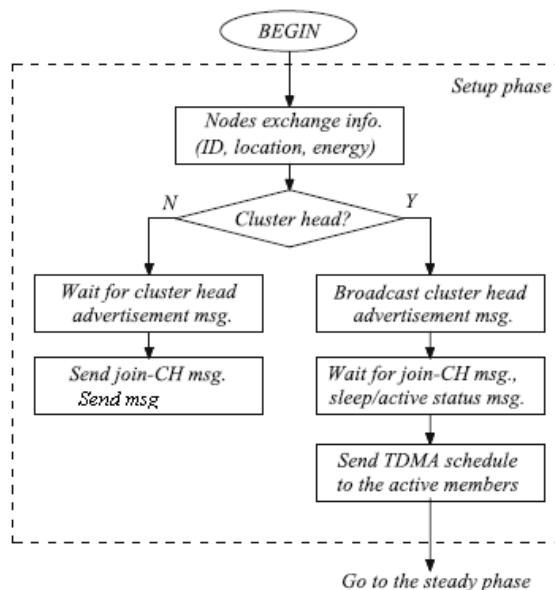
$$T(n) = \frac{P}{1 - P * (r \bmod \frac{1}{P})} \quad \text{if } n \in G$$

$$T(n) = 0 \text{ khác}$$
(1)

Hình 2.3 : Ngưỡng thiết lập Cluster Head

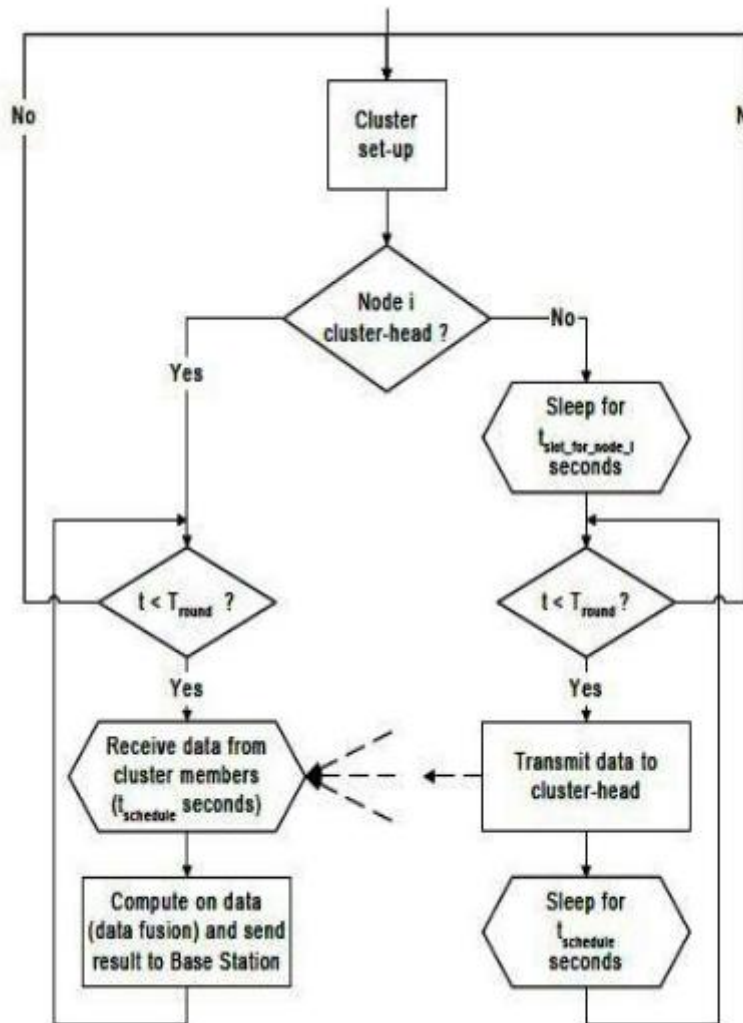
Ở đây P = tỷ lệ phần trăm mong muốn trở thành nút Cluster Head, r là chu kỳ hiện tại, và G là tập hợp các nút chưa được trở thành là Cluster Head trong 1/P chu kỳ. Bằng cách sử dụng so sánh với ngưỡng trên, mỗi nút sẽ được làm Cluster Head tại một số thời điểm trong vòng 1/P chu kỳ. Sau 1/P - 1 chu kỳ, T sẽ bằng 1 cho bất kỳ nút nào chưa được làm Cluster Head, và sau 1/P chu kỳ, tất cả các nút lại một lần nữa hội đủ điều kiện để trở thành Cluster Head.

Khi các nút trở thành Cluster Head bằng cách sử dụng công thức trên, các nút Cluster Head phải thông báo cho tất cả các nút khác trong mạng rằng họ đã trở thành Cluster Head trong chu kỳ hiện tại. Để làm được điều này nút Cluster Head phát một gói tin quảng bá ( Broadcast ) bằng cách sử dụng CSMA.



Hình 2.4 : Lưu đồ thuật toán cài đặt LEACH(www.slideshare.net)

Trong giai đoạn ổn định trạng thái, các nút cảm biến bắt đầu cảm biến và truyền phát số liệu về các Cluster Head. Các nút Cluster Head sau khi thu nhận tất cả các số liệu, xử lý dữ liệu rồi gửi lên nút chủ SINK. Sau một thời gian nhất định do nút chủ SINK quy định thì tất cả các nút sẽ bắt đầu vòng lặp mới. [4] [5]



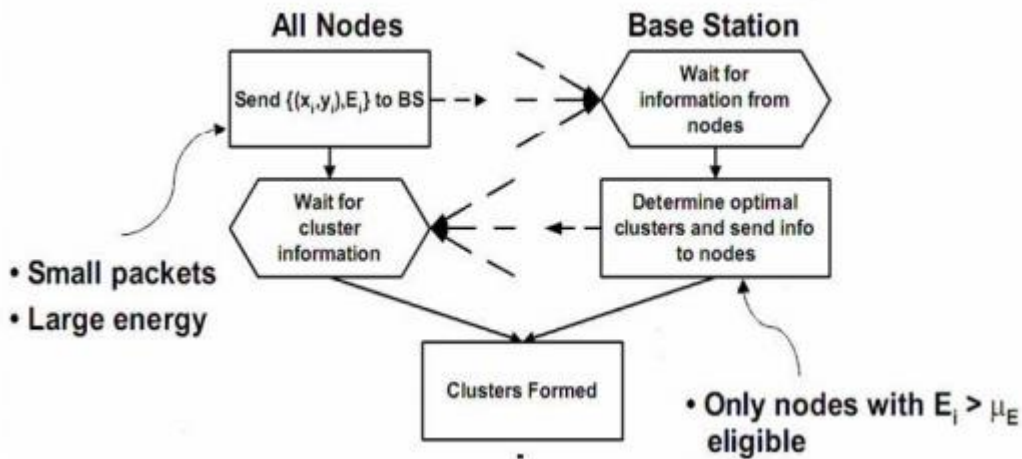
Hình 2.5 : Lưu đồ thuật toán giai đoạn ổn định LEACH(www.slideshare.net)

### 2.3 Giao thức cải tiến LEACH-C:

Trong mục trước LEACH đã được mô tả chi tiết, giao thức LEACH đã hình thành thuật toán phân cụm cho mạng, mỗi nút đều tự quyết định, lựa chọn phân cụm và chức năng cho mình. Điều này khiến cho giao thức không đảm bảo về số lượng các nút trong một cụm và số lượng các nút trở thành Cluster Head. Như vậy nếu có một cụm quá nhiều nút hoặc quá ít các nút Cluster Head sẽ ảnh hưởng rất nhiều đến hiệu suất tổng thể của hệ thống. Tuy nhiên nếu để nút chủ tự phân định nút nào là Cluster Head thì sẽ tránh được nhiều vấn đề mà LEACH



đang gặp phải. Đây là cơ sở cho LEACH-C (LEACH-Centralized), một giao thức mà sử dụng một thuật toán cụm tập trung và ổn định cùng một giao thức như LEACH (các nút gửi dữ liệu của họ vào các cluster-head, và các cluster-head tập hợp dữ liệu và gửi các tín hiệu tổng hợp cho các trạm cơ sở).



Hình 2.6 : Giai đoạn cài đặt của LEACH-C ([www.slideshare.net](http://www.slideshare.net))

Trong quá trình thiết lập các giai đoạn LEACH-C, mỗi nút gửi thông tin về vị trí hiện tại của nó và năng lượng cho nút chủ SINK. Nút chủ SINK sẽ xác định và chia cụm cho các nút trong hệ thống. Các cụm thành lập bởi nút chủ SINK nói chung sẽ tốt hơn so với những hình thành bằng cách sử dụng các thuật toán phân phối ngẫu nhiên. Sau khi hoàn thành bước cài đặt thì thuật toán LEACH-C cũng tiến hành bước ổn định giống với LEACH. [4] [5]

## 2.4 ZigBee:

- Giới thiệu về ZigBee:

ZigBee là một giao thức mạng không dây được dùng để kết nối các thiết bị với nhau. Công nghệ ZigBee được xây dựng dựa trên tiêu chuẩn 802.15.4 IEEE. Tiêu chuẩn này sử dụng tín hiệu radio có tần số ngắn và có cấu trúc hai tầng là tầng vật lý và tầng MAC. Công nghệ ZigBee dùng sóng radio và cũng có cấu trúc hai tầng như trên.

Tên gọi ZigBee lấy cảm hứng từ điệu nhảy theo đường zig-zag của ong mật (honey bee), điệu nhảy này được loài ong sử dụng để trao đổi thông tin với nhau về vị trí của hoa và nguồn nước. [2] [3]

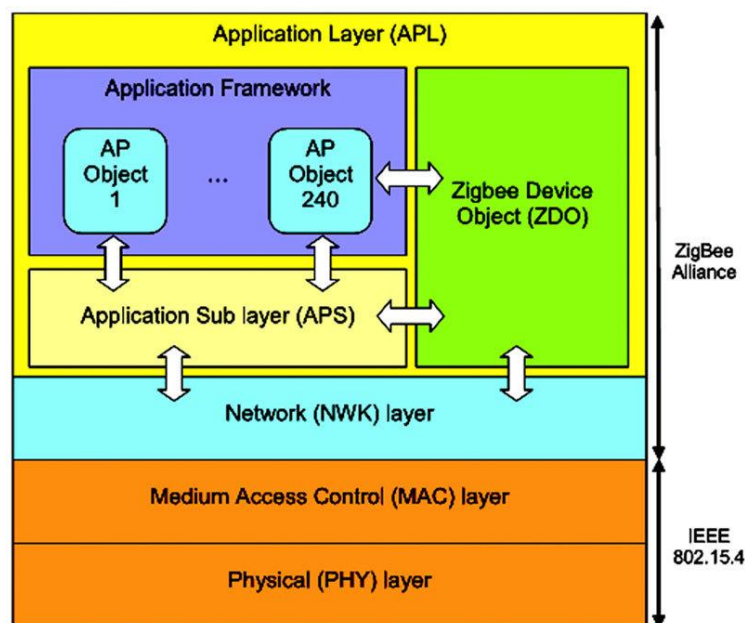
- Chuẩn truyền thông không dây IEEE 802.15.4:

IEEE (Institute of Electrical and Electronics Engineers) là tổ chức nghiên cứu phát triển các công nghệ liên quan đến thiết bị điện và điện tử. Nhóm 802 của tổ chức chuyên nghiên cứu về công nghệ mạng, bộ phận 802.15 được dành riêng cho chuẩn mạng không dây. Chuẩn IEEE 802.15.4 quy định truyền thông trên sóng radio trong phạm vi 10 mét đến 100 mét và hoạt động ở ba dải tần chính:

- \* Dải 868 - 868.8 MHz: chỉ một kênh tín hiệu, trong dải này tốc độ truyền là 20kb/s.
- \* Dải 902 - 928 MHz: có 10 kênh tín hiệu từ 1 - 10 với tốc độ truyền thường là 40kb/s.
- \* Dải 2.4 - 2.4835 GHz: 16 kênh tín hiệu từ 11 - 26 với tốc độ truyền 250 kb/s.

- Cấu trúc của giao thức ZigBee:

Giao thức ZigBee có cấu trúc ngăn xếp nhiều tầng, trong đó tầng vật lý và tầng MAC được định nghĩa giống chuẩn IEEE 802.15.4. Sau đó ZigBee Alliance đã xây dựng thêm 4 thành phần chính: tầng mạng, tầng ứng dụng, đối tượng thiết bị ZigBee (ZigBee device objects – ZDO) và các đối tượng người dùng .



Hình 2.7 : Cấu trúc của mạng ZigBee(adlt.com.au)

ZOD là cải tiến đáng kể nhất của ZigBee, đây là đối tượng thực hiện những tác vụ quan trọng như định nghĩa vai trò của thiết bị, truy cập mạng, bảo mật thiết bị,..

- *Thành phần mạng ZigBee:*

Một mạng kiểu ZigBee gồm có 3 loại thiết bị:

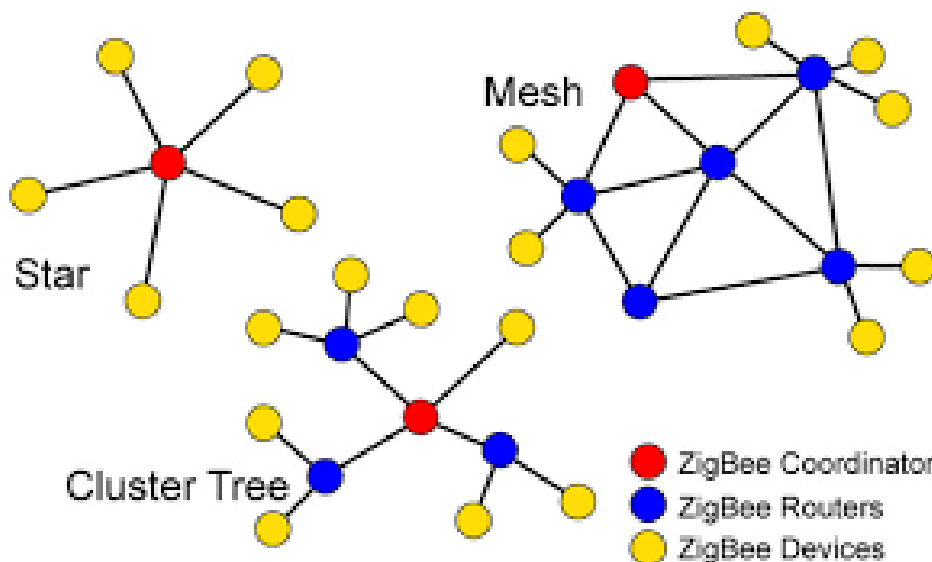
\* ZC (Zigbee Coordinator): Đây là nút chủ của một mạng ZigBee. Một mạng ZigBee có duy nhất một nút chủ và ZigBee sử dụng nút chủ để xây dựng, thiết lập, cấu hình mạng.

\* ZR (Zigbee Router): Nút Router có nhiệm vụ truyền tải thông tin từ nút thành phần đến nút chủ ZC.

\* ZED (Zigbee End Device): Nút này được xem là thiết bị đầu cuối, nút chuyên thu thập dữ liệu và gửi trực tiếp lên nút chủ ZC hoặc thông qua nút Router.

- *Mô hình mạng ZigBee:*

Mô hình mạng ZigBee có ba dạng chính: dạng hình sao, dạng lưới và dạng cây.



Hình 2.8 : Mô hình mạng ZigBee (<http://arduino.vn/>)

Hình sao (Star network): Mạng hình sao không sử dụng các nút Router mà thay vào đó các nút thành phần liên hệ trực tiếp đến nút chủ ZC. Mô hình mạng ngang hàng đã được nhắc đến ở trên.

Hình lưới (Mesh network): Mô hình mạng lưới có các nút liên kết chặt chẽ với nhau. Trong mô hình này dữ liệu có thể vận chuyển qua nhiều khâu trung gian ZR để đến được với nút chủ ZC. Điều này phụ thuộc vào cấu hình cũng như cài đặt hệ thống của người xây dựng.

Hình cây (Cluster network): Mạng hình cây hoạt động theo cách các dữ liệu từ nút thành phần sẽ phải thông qua nút Router để đến được với nút chủ ZC. Mô hình mạng này dễ cài đặt và tính mở rộng cao nên được rất nhiều người tin dùng. [2] [3]

## Chương III. XÂY DỰNG MẠNG CẢM BIẾN DỰA TRÊN LEACH VÀ ZIGBEE

### 3.1. Đặt vấn đề:

Mạng cảm biến không dây mang lại hiệu quả rất lớn trên nhiều lĩnh vực trong cuộc sống và trong nghiên cứu. Hiện nay có rất nhiều chuyên gia nghiên cứu và đưa ra nhiều giao thức mạng cảm biến không dây. ZigBee là mô hình mạng cảm biến có rất nhiều ưu điểm cho nên ZigBee được người dùng lựa chọn để xây dựng mạng cảm biến không dây. Bên cạnh những lợi thế to lớn của ZigBee như đã phân tích ở phần trước thì ZigBee còn có một vài khuyết điểm đi cùng. Trong đó khuyết điểm lớn nhất của ZigBee là thời gian sống của mạng. Mô hình mạng ZigBee gồm có ba thành phần chính là nút Coordinator, nút Router và nút End Device. Bình thường nút Coordinator sẽ được đặt ở một vị trí cố định và sử dụng nguồn điện ổn định từ mạng lưới điện. Nút Router và nút End Device sẽ sử dụng nguồn điện cung cấp từ năng lượng tích trữ như pin hoặc năng lượng mặt trời. Đặc điểm nút End Device sẽ thu thập dữ liệu và gửi dữ liệu đến nút Router theo từng chu kỳ, nút Router sẽ chuyển tiếp dữ liệu lên nút Coordinator. Sau khi nút End Device gửi dữ liệu đi thì có thể được người thiết kế cho ngủ đông để tiết kiệm năng lượng và chờ chu kỳ tiếp theo. Ngược lại nút Router sẽ phải hoạt động liên tục để lắng nghe dữ liệu từ nhiều nút End Device gửi về. Cho nên nút Router sẽ tiêu tốn nhiều năng lượng nhất. Sau một thời gian hoạt động nút Router sẽ tiêu thụ hết lượng năng lượng từ pin dẫn đến việc mạng sẽ không hoạt động được nữa. Trong khi đó mức năng lượng còn trên các nút End Device vẫn còn nhiều.

LEACH là giao thức mạng cảm biến không dây với thế mạnh kéo dài sự sống của mạng. LEACH sử dụng phương pháp chia đều công việc cho toàn bộ các nút trong cụm. Trên lý thuyết mức năng lượng tiêu thụ ở các nút tương đương nhau nên khi hoạt động các nút sẽ hết năng lượng gần như đồng thời. Như vậy sự sống của mạng cảm biến không dây sẽ được tối ưu hóa. Bên cạnh lợi thế trên thì trong thực tế khoảng cách, vị trí địa lý của các nút thành phần là một vấn đề giao thức LEACH gặp phải. Khoảng cách, vị trí địa lý ảnh hưởng

trực tiếp đến hiệu suất và mức tiêu thụ điện năng của thiết bị truyền thông. LEACH sử dụng một hàm ngẫu nhiên để chọn nút Cluster Head nên khi chọn nút Cluster Head thì nút thành phần có thể chọn một nút Cluster Head ở xa hơn thay vì chọn một nút Cluster Head ở gần. Điều này dẫn đến khi mạng hoạt động một thời gian sẽ xuất hiện những nút thành phần hết năng lượng trước dẫn đến hiệu quả của mạng chưa cao.

Trong thời gian tìm hiểu và xây dựng đề tài này tôi đã gặp phải hai vấn đề cơ bản như sau:

- Kéo dài sự sống của mạng cảm biến ZigBee
- Tối ưu hóa việc chia cụm và tiết kiệm năng lượng cho giao thức LEACH

### **3.2. Giải pháp đề xuất:**

Vấn đề đầu tiên là kéo dài sự sống của mạng cảm biến ZigBee tôi đề xuất áp dụng giao thức LEACH vào trong ZigBee. Hệ thống vẫn sử dụng chuẩn truyền thông không dây IEEE 802.15.4 và cấu trúc mạng của ZigBee nhưng việc đóng gói dữ liệu, xác định đường truyền dữ liệu lại theo giao thức LEACH. Như đã giới thiệu ở phần trước về giao thức LEACH, giao thức LEACH bao gồm ba thành phần chính là: Nút chủ, nút Cluster Head và nút Non Cluster Head hoặc gọi là nút lá. Khi áp dụng giao thức LEACH vào trong ZigBee thì nút chủ cũng chính là nút Coordinator trong Zigbee, nút ClusterHead cũng hoạt động giống với nút Router và nút lá cũng có chức năng tương tự với nút End Device trong Zigbee. Các nút lá thay phiên nhau đảm nhận trở thành nút Cluster Head, các nút không phải hoặc đã từng là Cluster Head sẽ lựa chọn một trong những Cluster Head xung quanh để truyền dữ liệu lên. Nút Cluster Head hình thành cụm và chịu trách nhiệm truyền dữ liệu từ các nút thành phần của mình lên nút chủ (SINK). Như đã trình bày ở trên về yếu điểm của mạng cảm biến ZigBee là do nút Router hoạt động quá nhiều dẫn đến việc hết năng lượng trước các nút khác. Khi áp dụng giao LEACH vào trong các nút sẽ thay phiên nhau chịu trách nhiệm đó cho nên mức năng lượng tiêu hao sẽ chia đều cho toàn bộ các nút trong hệ thống. Như vậy hệ thống sẽ hoạt động được lâu hơn và đã giải quyết được vấn đề nêu trên. [8]

Vấn đề thứ hai là tối ưu hóa việc chia cụm và tiết kiệm năng lượng cho giao thức LEACH. Do việc lựa chọn ngẫu nhiên các nút Cluster Head nên việc phân cụm gặp nhiều trường hợp không hợp lý. Giao thức LEACH-C là giao thức cải tiến LEACH, trong đó việc lựa chọn các nút trở thành Cluster Head dựa trên mức tiêu thụ điện năng của các nút thành phần. Những nút còn nhiều điện năng hơn sẽ được ưu tiên lựa chọn trở thành Cluster Head và công việc này được nút chủ (SINK) thực hiện. Như đã nhắc đến ở phần trên về việc đo mức năng lượng tiêu thụ điện năng khá khó khăn và phải gắn thêm thiết bị ngoại vi nên tôi không áp dụng LEACH-C vào trong ZigBee. Trong đề tài này tôi đề xuất sử dụng hiệu suất truyền nhận gói tin thay vì mức độ tiêu thụ năng lượng của thiết bị. Thiết bị truyền thông không dây có hiệu suất truyền nhận gói tin phụ thuộc vào nhiều yếu tố như: Khoảng cách, vật cản, môi trường, chất lượng thiết bị,... Khi trong mạng có hai nút ClusterHead trở lên, từ một nút lá đang xét sẽ lựa chọn một trong những ClusterHead trên để truyền dữ liệu đến. Nút lá đang xét sẽ so sánh hiệu suất truyền dữ liệu giữa nó và các nút ClusterHead trên. Nút có hiệu suất truyền dữ liệu cao nhất sẽ được nhận là Cluster Head và thực hiện việc truyền dữ liệu. Hiệu suất truyền dữ liệu cao chứng tỏ điều kiện thuận lợi giữa hai nút với nhau. Với những yếu tố ảnh hưởng đến hiệu suất truyền dữ liệu đã phân tích ở trên sẽ chứng minh được việc tiêu thụ năng lượng trên thiết bị tiết kiệm hơn, hiệu quả hơn.

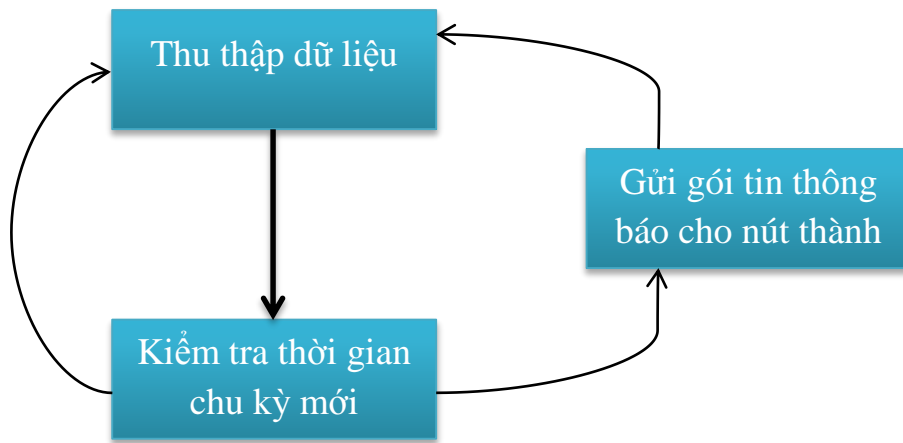
### **3.3. Cách thức triển khai giải pháp:**

Để triển khai được giải pháp trên chúng ta có hai bước quan trọng. Bước đầu tiên là xây dựng một hệ thống sử dụng mạng ZigBee đơn thuần. Bước thứ hai là áp dụng giao thức LEACH vào trong mạng cảm biến vừa xây dựng. Để xây dựng được mạng ZigBee chúng ta phải có thiết bị truyền thông và bo mạch điều khiển. Xbee là thiết bị truyền thông được sử dụng nhiều hiện nay. Xbee được nhà sản xuất tích hợp mạng ZigBee vào trong nên việc cài đặt hệ thống mạng ZigBee rất dễ dàng và tiện lợi. Với nhiều ưu điểm của Xbee sẽ được tôi giới thiệu ở phần sau nên tôi chọn Xbee là thiết bị truyền thông không dây trong đề tài của mình. Bo mạch chủ tôi chọn Arduino Nano, Arduino Nano là phiên

bản rút gọn của Arduino nhưng cũng đủ để đáp ứng được những yêu cầu trong đề tài này. Việc cài đặt và kết nối Arduino cho Xbee để hệ thống có thể hoạt động được tôi sẽ trình bày chi tiết ở chương sau. Xây dựng mạng cảm biến không dây ZigBee cơ bản và đặt ra các điều kiện cụ thể để phân tích lợi thế, yếu điểm của ZigBee như: Theo dõi mạng hoạt động trong thời gian dài, tần suất hoạt động cao, lưu lượng thông tin truyền trong mạng lớn, vị trí địa lý các nút không thuận lợi,... sau đó rút ra những yếu điểm cơ bản đã nêu ở trên như: Thời gian sống không dài, cách thức truyền tin khá lãng phí,...

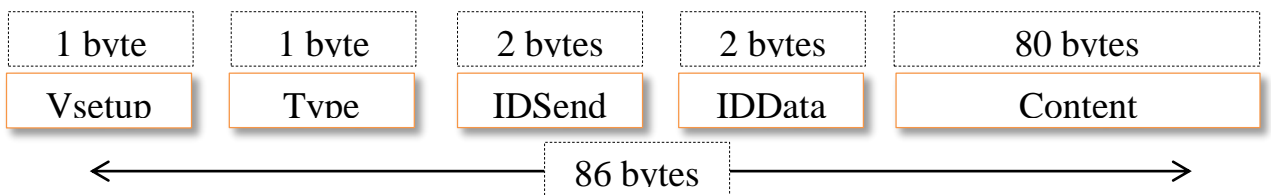
Sau khi xây dựng xong hệ thống Xbee thuần thì chúng ta bắt đầu áp dụng giao thức LEACH vào trong ZigBee để khắc phục hai vấn đề gặp phải đã nêu ở trên. Thiết bị Xbee được cài đặt mô hình ZigBee sẽ sử dụng phương thức truyền thông gói tin ở dạng broadcast. Gói tin sẽ được phát ra xung quanh cho nên muốn áp dụng giao thức LEACH thì phải định tuyến chính xác đường đi của gói tin trong mạng. Xbee cung cấp API truyền tin và công việc này được xác lập trên bo mạch Arduino kết nối với Xbee đó. Khi áp dụng giao thức LEACH vào trong ZigBee thì nút Router trong ZigBee được bỏ đi. Các nút thành phần sẽ được tích hợp hai chức năng, một là thay phiên nhau đảm nhận công việc truyền tải dữ liệu hai là trực tiếp thu thập dữ liệu và gửi về nút chủ cụm (Cluster Head). Như vậy trong hệ thống khi xây dựng sẽ có hai thành phần chính là nút chủ SINK, nút thành phần. Giao thức LEACH hoạt động theo vòng tuần hoàn và trong vòng tuần hoàn có nhiều chu kỳ. Bình thường số chu kỳ sẽ được người thiết kế chỉ định và chính là số nút thành phần có trong mạng. Chu kỳ chính là thời gian một nút thành phần có thể đảm nhận công việc Cluster Head trong vòng tuần hoàn đó. Thời gian này được nút chủ SINK quy định. Nút chủ SINK sẽ đảm nhận công việc thu thập dữ liệu từ các nút Cluster Head và gửi thông báo đến tất cả các nút thành phần khi chu kỳ mới bắt đầu.





Hình 3.1. Sơ đồ chức năng của nút chủ SINK

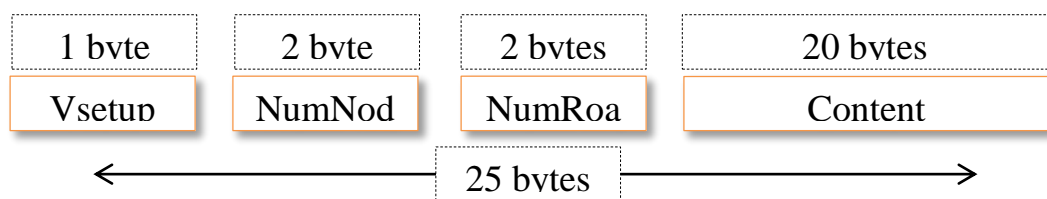
Nút chủ SINK chỉ nhận dữ liệu từ các nút Cluster Head trong hệ thống mạng cảm biến. Cho nên khi các nút thành phần trở thành nút Cluster Head thì phải chuyển tiếp dữ liệu từ các nút khác trong cụm lên cho nút SINK và tự thu thập dữ liệu gửi trực tiếp lên nút SINK. Như vậy dữ liệu nút SINK thu thập được sẽ có hai dạng. Dạng thứ nhất là dữ liệu được chuyển tiếp và dạng thứ hai là dữ liệu riêng của nút Cluster Head.



Hình 3.2. Khung dữ liệu nút SINK nhận được

Trong đó Vsetup là biết dạng int chứa đựng chức năng của gói tin. Trường này dùng để xác thực gói tin và phân biệt những gói tin khác nhau gửi đến. Với Vsetup bằng 5 thì dữ liệu chính là được gửi về từ các nút Cluster Head. Type là biến lưu dạng gói tin đến, như đã phân tích ở trên có hai dạng dữ liệu từ một nút Cluster Head gửi đến cho nút SINK. Với type bằng 0 tức là dữ liệu cá nhân của Cluster Head, type bằng 1 là dữ liệu được nút Cluster Head chuyển tiếp. IDSend là định danh của Cluster Head, IDData chính là định danh của nút thành phần thu thập dữ liệu. Content là nội dung dữ liệu mà các nút thu thập được qua hệ thống cảm biến của mình. Khi nhận được dữ liệu nút SINK sẽ sử dụng nhiều phương pháp để xử lý dữ liệu như: hiển thị qua monitor cho người thiết kế, truyền lên internet, truyền đến ứng dụng trong máy tính qua cổng COM,...

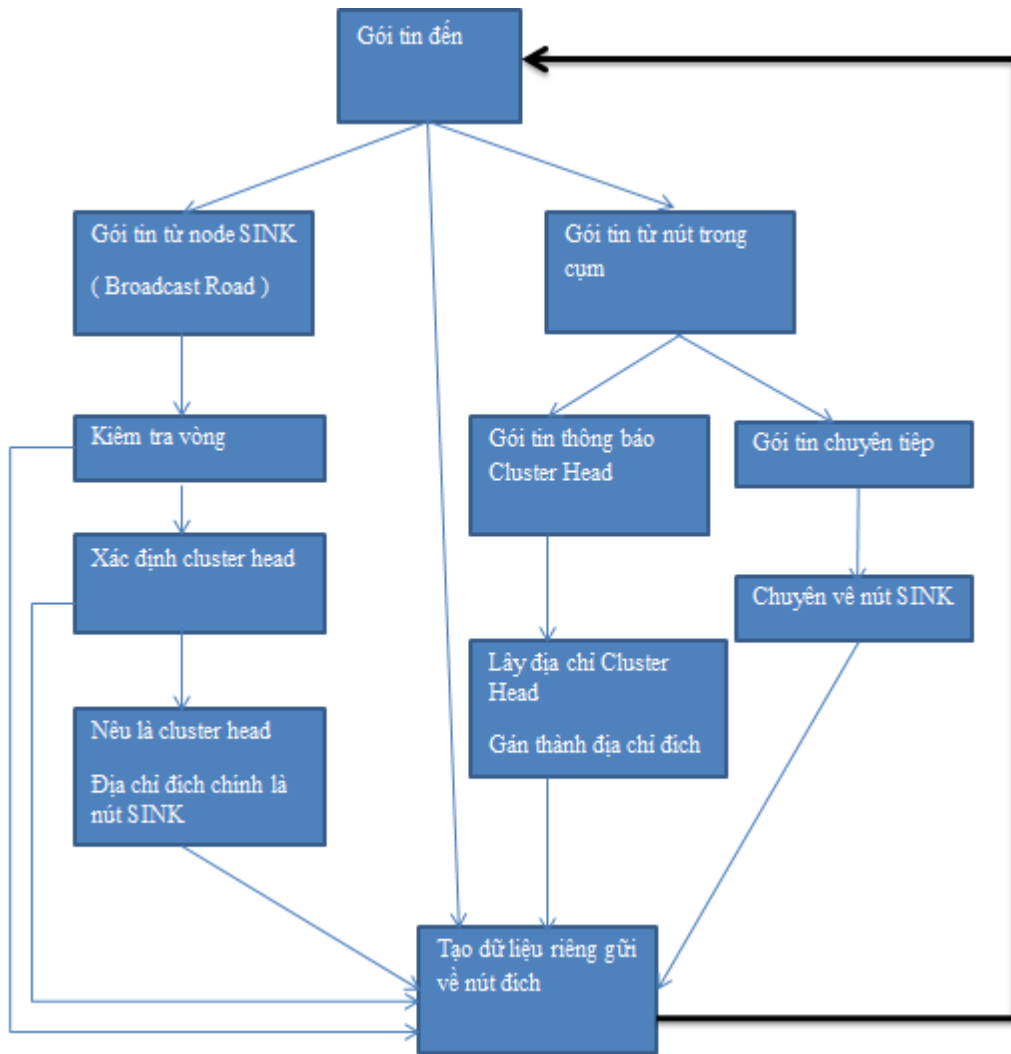
Xử lý dữ liệu xong nút SINK sẽ kiểm tra thời gian của chu kỳ. Trong lập trình Arduino có cung cấp cho người dùng một biến số millis, biến số này lưu trữ thời gian từ khi Arduino được cấp nguồn và hoạt động. Biến millis sẽ được đặt lại mặc định là 0 khi Arduino mất điện. Kiểm tra biến millis để thiết lập thời gian cho chu kỳ. Trong đề tài này tôi chọn chu kỳ 20 phút, khi bắt đầu hoạt động hoặc bắt đầu chu kỳ mới nút SINK sẽ gửi dữ liệu đi với dạng broadcast đến các nút thành phần trong hệ thống. Nếu chưa hết chu kỳ hiện tại thì nút SINK lại quay trở lại lắng nghe dữ liệu gửi đến từ các nút Cluster Head.



Hình 3.3. Khung dữ liệu nút SINK gửi đi.

Trong đó Vsetup là biến chức năng gói tin, Vsetup ở đây tôi cố định bằng 1 để phân biệt với các gói tin khác. NumNode là biến lưu trữ số các nút thành phần có trong hệ thống mạng cảm biến. NumRoad là biến lưu chu kỳ hiện tại của hệ thống. Tôi xây dựng số chu kỳ chính bằng số các nút trong hệ thống cho nên khi chu kỳ lớn hơn số các nút thì biến NumRoad sẽ được đặt lại bằng 1. Content là nội dung gói tin, tôi thêm trường content vào để giúp tôi theo dõi được hệ thống. Khi hệ thống hoạt động tốt có thể không cần đến biến content này nữa.

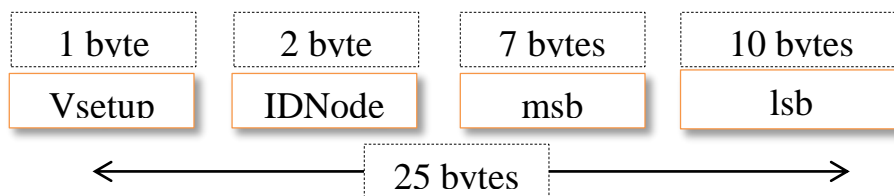
Thành phần quan trọng và khá phức tạp của hệ thống chính là nút thành phần. Chức năng của nút thành phần được thể hiện ở sơ đồ dưới.



Hình 3.4. Sơ đồ chức năng nút thành phần

Mỗi Arduino Nano sẽ thực hiện một vòng lặp tuần hoàn trong suốt thời gian hoạt động. Theo sơ đồ hoạt động được xây dựng như trên thì bước đầu vòng lặp Arduino Nano sẽ kiểm tra thông tin đến. Dựa vào trường đầu tiên của gói tin ta có ba loại gói tin thông hành trong hệ thống. Một là gói tin cài đặt được nút SINK phát broadcast trong mạng, hai là gói tin thông báo đích danh cluster head và gói tin dữ liệu riêng của nút trong mạng gửi đến. Do đặc thù Xbee nên mỗi một vòng lặp chỉ nhận được một gói tin. Với gói tin broadcast đến từ nút SINK tức là vòng lựa chọn cluster head mới đã bắt đầu. Khi nhận được gói tin gồm 25 bytes đã trình bày ở trên nút thành phần bắt đầu kiểm tra biến NumRoad. Nếu biến NumrRoad bằng 1 thì hệ thống bắt đầu một vòng tuần hoàn mới khi đó tất cả các nút thành phần được thiết đặt lại từ đầu. Nếu biến NumRoad khác 1 thì nút thành phần kiểm tra xem ở những chu kỳ trước mình

đã trở thành Cluster Head chưa. Nếu đã từng trở thành Cluster Head thì bỏ qua và xuống bước tạo dữ liệu riêng rồi gửi đến nút đích. Nút đích được thiết lập mặc định là địa chỉ của nút SINK, địa chỉ này thay đổi khi nút thành phần này tham gia vào một cụm các nút có Cluster Head quản lý và trở lại mặc định khi chính nó trở thành Cluster Head. Nếu nút thành phần này chưa từng trở thành Cluster Head thì sẽ sang bước tiếp theo là xác định Cluster Head. Các nút sẽ tự xác định xem mình có trở thành Cluster Head hay không bằng công thức LEACH đã trình bày ở phần trước. Nếu không phải Cluster Head thì nút thành phần lại đóng gói dữ liệu riêng và gửi đến nút đích. Nếu trở thành nút Cluster Head thì sẽ gửi thông tin quảng bá BroadCast đến các nút khác để thông báo mình trở thành Cluster Head. Qua đó để thành lập cụm các nút do mình quản lý.

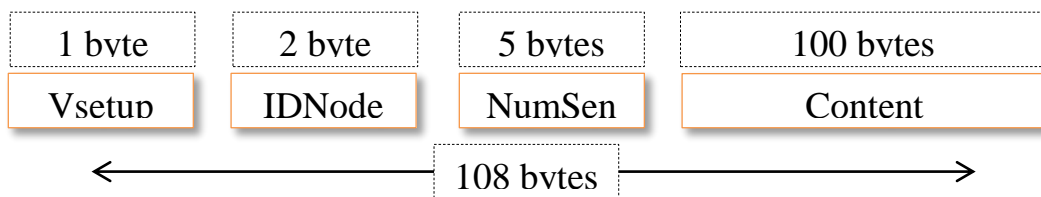


Hình 3.5. Khung dữ liệu Broad Cast của Cluster Head

Trong đó Vsetup có giá trị bằng 2 dùng để chỉ chức năng của gói tin. IDNode là định danh, msb và lsb là địa chỉ của nút Cluster Head đó.

Nếu nhận được gói tin từ nút Cluster Head khác, nút thành phần sẽ kiểm tra trong chu kỳ hiện tại mình đã tham gia vào cụm nào chưa. Nếu đã tham gia thì bỏ qua bước này, nếu chưa sẽ chọn ngẫu nhiên hoặc dựa vào hiệu suất truyền dữ liệu để lựa chọn Cluster Head cho mình.

Nếu nhận được gói tin từ nút thành phần khác chuyển đến thì có nghĩa mình đang là Cluster Head.



Hình 3.6. Khung dữ liệu được gửi từ nút thành phần đến Cluster Head

Ở đây Vsetup bằng 3, IDNode là định danh của nút gửi dữ liệu đến và Content là nội dung thu thập được từ các cảm biến của nút thành phần. Biến NumSend dùng để lưu trữ số gói tin đã gửi và khi chu kỳ mới bắt đầu tức là có nút ClusterHead mới thì biến sẽ trở lại giá trị mặc định là 0. Mỗi gói tin được gửi đi biến sẽ tăng lên một đơn vị.

Dữ liệu trong mỗi nút có các biến như sau :

- hieusuatX: lưu trữ thông tin hiệu suất truyền tải dữ liệu từ nó đến nút X. Nút X chính là định danh của các nút còn lại của hệ thống ngoại trừ nút SINK. Giá trị mặc định của hieusuatX là -1.
- DataRevX : lưu trữ thông tin số các gói tin nhận được từ các nút thành phần X gửi đến nó. Biến này sẽ trở lại giá trị mặc định là 0 khi nút hiện tại trở thành ClusterHead. Mỗi khi nhận được gói tin đến từ nút X biến sẽ tăng một đơn vị.

Mỗi lần nhận được gói tin đến từ nút X thì ClusterHead sẽ thay đổi giá trị của biến hieusuatX theo công thức sau :

$$\text{hieusuatX} = (\text{hieusuatX} + (\text{DataRevX} * 100 / \text{NumDataSend})) / 2 \quad (2)$$

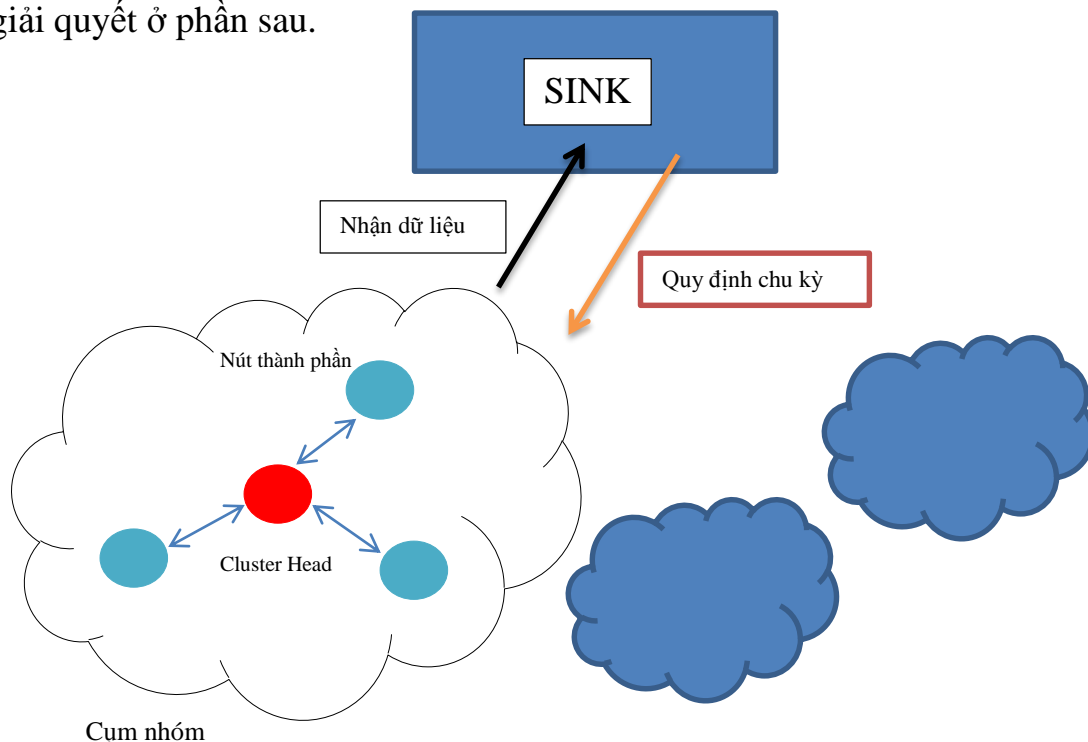
Như vậy ta luôn có giá trị trung bình hieusuatX để đánh giá hiệu suất truyền tin của hai nút. Giá trị này được nút lá dựa vào để lựa chọn nút ClusterHead thích hợp khi mạng phát sinh nhiều hơn một nút ClusterHead.

Sau một thời gian hoạt động thì biến hieusuatX sẽ được xác lập chính xác hơn dẫn đến việc lựa chọn Cluster Head quản lý mình đạt hiệu quả hơn.

## Chương IV. XÂY DỰNG HỆ THỐNG VÀ ĐÁNH GIÁ GIẢI PHÁP

### 4.1. Tổng quan hệ thống:

Một mô hình mạng ZigBee muốn đánh giá chất lượng truyền thông, sự ổn định, hiệu năng tiêu thụ,... cần rất nhiều nút, giao thức LEACH cũng vậy. Do điều kiện đề tài cho phép nên tôi chỉ sử dụng bảy thiết bị Xbee và bảy bo mạch Arduino để xây dựng hệ thống. Với mô hình như thế chúng ta phải truyền tải lượng thông tin lớn trên một khoảng thời gian nhất định để có thể bù đắp phần nào hạn chế thiết bị trong hệ thống. Việc này dẫn đến khá nhiều vấn đề cần được giải quyết ở phần sau.



Hình 4.1 Sơ đồ hệ thống

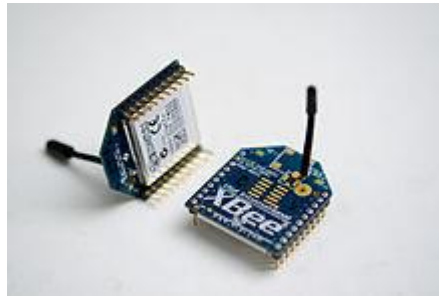
Hệ thống có bảy thiết bị truyền thông Xbee và bảy bo mạch Arduino Nano. Mỗi một Xbee được kết nối đến Arduino Nano hình thành một nút trong hệ thống. Vì đặc điểm chung của ZigBee và LEACH đều có một nút chủ hay còn gọi là nút SINK. Sáu nút còn lại được xây dựng thành hai cụm theo phương thức LEACH.

### 4.2. Các thiết bị phần cứng:

#### 4.2.1. Thiết bị truyền thông Xbee:

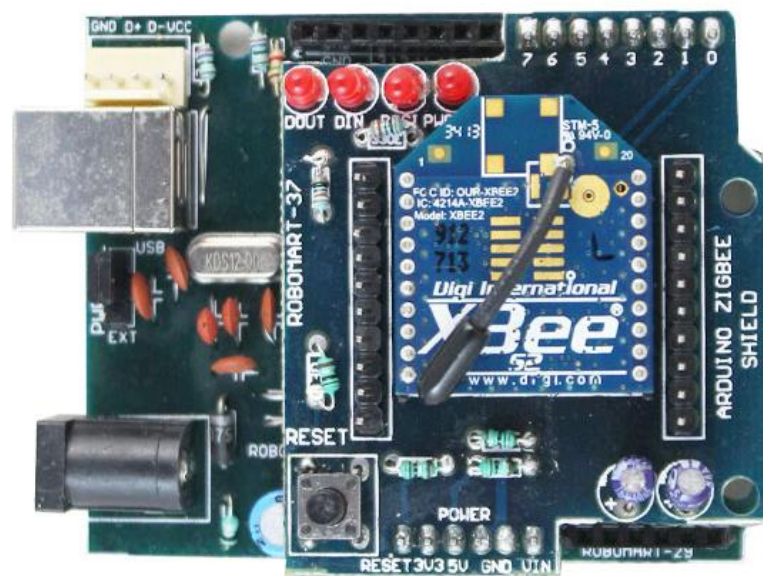
Digi Xbee là tên thương hiệu của một nhóm các module radio tương thích với các yêu tố hình thức của Digi International. Các radia Xbee đầu tiên đã được

giới thiệu dưới nhãn hiệu MaxStream năm 2005 và dựa trên tiêu chuẩn IEEE 802.15.4-2003.



Hình 4.2 : Thiết bị Xbee (<http://arduino.vn/>)

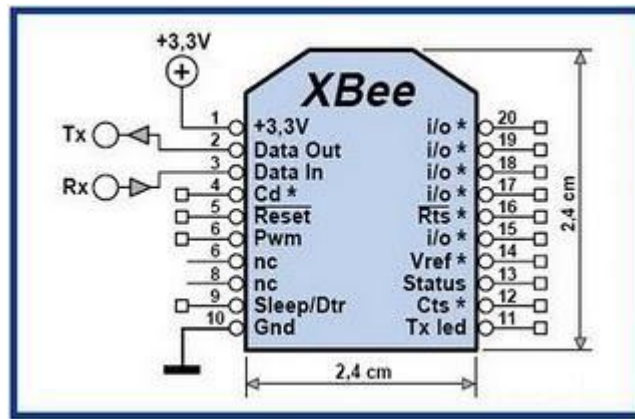
Xbee là thiết bị truyền thông không dây được ưa chuộng trong lĩnh vực mạng truyền thông không dây. Qua rất nhiều phiên bản được nâng cấp và cải tiến thì hiện nay Xbee S2C được sử dụng khá rộng rãi trên nhiều quốc gia và nhiều lĩnh vực trong cuộc sống. Với Xbee ta có thể sử dụng phần mềm chuyên dụng như XCTU để cài đặt. Nhà sản xuất đã hỗ trợ người dùng nhiều mô hình mạng có sẵn, nhưng chủ yếu Xbee hoạt động theo mạng cảm biến ZigBee. Với các thư viện lập trình sẵn cho phép Xbee kết nối đến những bo mạch thông dụng như Arduino, Raspberry, ... Với những ứng dụng đặc biệt thì Xbee có thể hoạt động độc lập nhưng phần lớn người dùng vẫn sử dụng phương thức kết hợp để có thể linh động trong việc truyền tin và đa dạng thông tin trên kênh truyền.



Hình 4.3 : Bộ kết hợp của Xbee và Arduino (<http://arduino.vn/>)

Xbee sử dụng nguồn 3.3V, trong đề tài này tôi sử dụng năm chân cơ bản của Xbee là chân nguồn, chân Tx, chân Rx, chân Sleep, chân GND





Hình 4.4 : Cấu trúc Xbee (<http://arduino.vn/>)

#### 4.2.2. Bo mạch Arduino Nano:

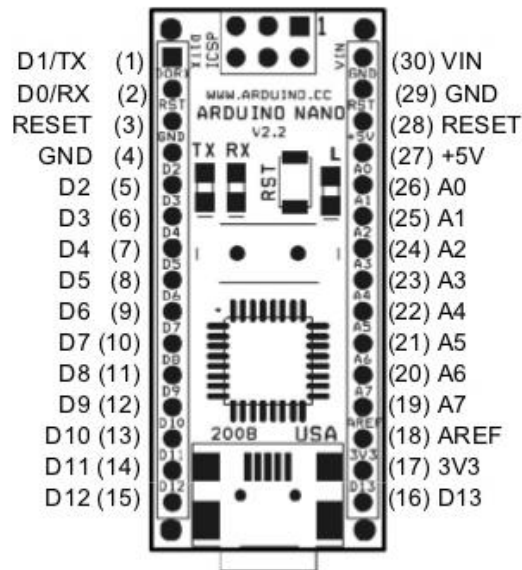
Arduino là một board mạch vi xử lý, nhằm xây dựng các ứng dụng tương tác với nhau hoặc với môi trường được thuận lợi hơn. Phần cứng bao gồm một board mạch nguồn mở được thiết kế trên nền tảng vi xử lý AVR Atmel 8bit, hoặc ARM Atmel 32-bit. Những Model hiện tại được trang bị gồm 1 cổng giao tiếp USB, 6 chân đầu vào analog, 14 chân I/O kỹ thuật số tương thích với nhiều board mở rộng khác nhau.



Hình 4.5 : Arduino Nano (<http://arduino.vn/>)

Arduino Nano là một phiên bản được rút gọn của Arduino. Arduino Nano nhỏ gọn hơn nhiều so với các bản Arduino khác và các chân được thiết kế được gắn trên bo mạch giúp người dùng có thể cắm Arduino trực tiếp trên các breadboard.



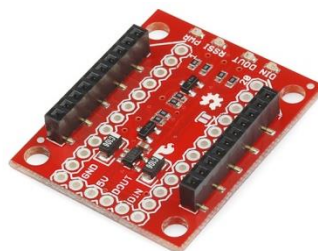


Pin No.	Name	Type	Description
1-2, 5-16	D0-D13	I/O	Digital input/output port 0 to 13
3, 28	RESET	Input	Reset (active low)
4, 29	GND	PWR	Supply ground
17	3V3	Output	+3.3V output (from FTDI)
18	AREF	Input	ADC reference
19-26	A7-A0	Input	Analog input channel 0 to 7
27	+5V	Output or Input	+5V output (from on-board regulator) or +5V (input from external power supply)
30	VIN	PWR	Supply voltage

Hình 4.6: Sơ đồ cấu trúc Arduino Nano (<http://arduino.vn/>)

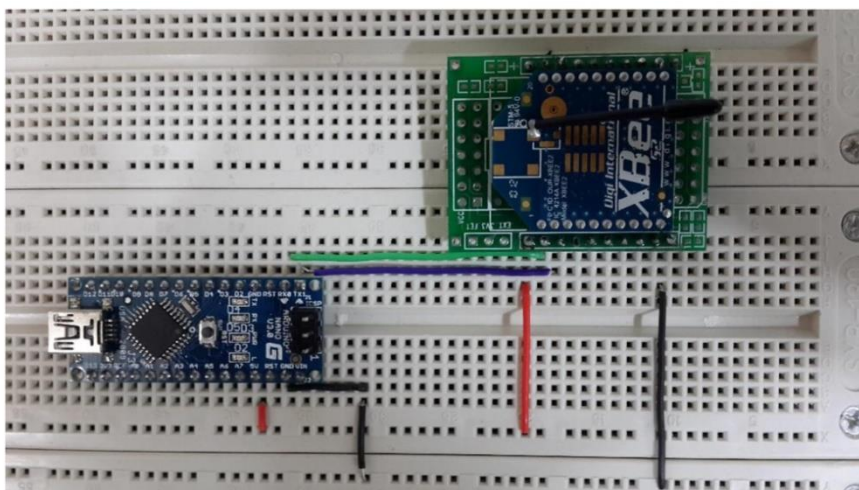
#### 4.2.3. Cài đặt hệ thống cơ bản

Xbee được thiết kế với chân cắm dùng chuẩn 2.0mm cho nên để thuận tiện kết nối với Arduino Nano ta cần phải có đế cắm riêng với chuẩn 2.5mm cho Xbee. Khi đó ta có thể dùng dây cáp thông thường để tiến hành kết nối Xbee và Arduino Nano.



Hình 4.7 : Đế chuyển đổi từ 2.0 sang 2.5 (<http://arduino.vn/>)

Sau khi cắm Xbee vào đế chuyển đổi chân ta có thể dùng dây kết nối hoặc dùng Bread board để tiện lợi và gọn gàng hơn như sau.



*Hình 4.8 : Xbee và Arduino Nano trên Bread Board*

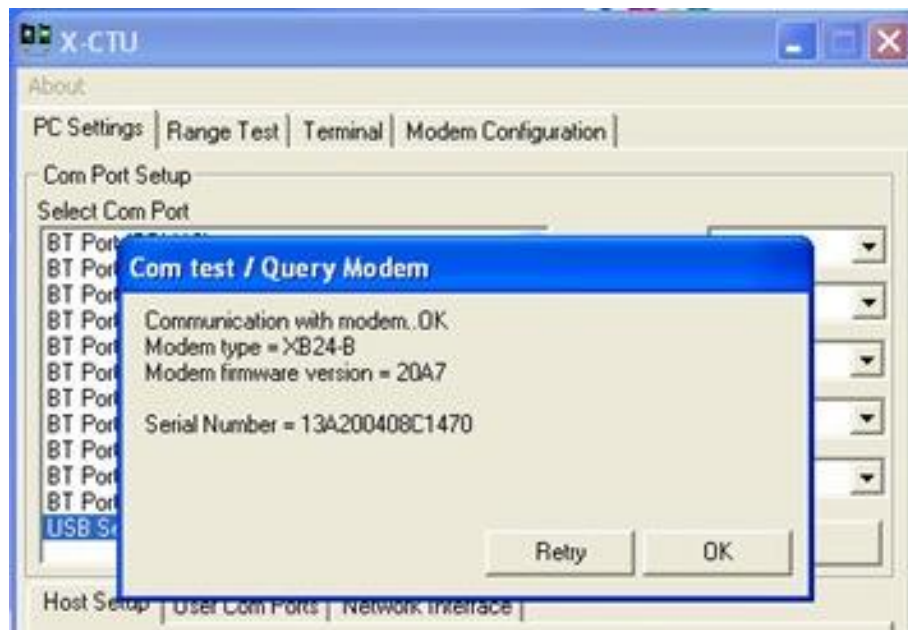
Bước đầu tiên phải làm là dùng Xbee và Arduino để tạo ra một mạng cảm biến sử dụng mạng ZigBee cơ bản. Hệ thống mạng phải hoạt động bình thường và có thể truyền tải khối lượng thông tin trên đường truyền. Có rất nhiều tài liệu hướng dẫn cách cài đặt và kết nối một mạng cảm biến Xbee như trên. Dưới đây là một hướng dẫn cơ bản:

Muốn có một mạng Xbee đơn thuần thì ban đầu phải cài đặt cấu hình cho các thiết bị Xbee. Thiết bị Xbee được kết nối đến máy tính qua phần mềm XCTU và một bo mạch kết nối được thiết kế riêng cho Xbee. Trong đề tài này tôi sử dụng một Xbee làm nút chủ và bốn Xbee khác được cấu hình là ZED (Zigbee End Device).



*Hình 4.9 : Bộ cấu hình cho Xbee*

Tiến hành cắm Xbee vào bộ cấu hình và kết nối cùng máy tính qua cổng USB. Việc kết nối qua cáp USB nhưng khi chương trình XCTU sẽ nhận đó là một cổng COM ảo. Sau đó bật chương trình XCTU lên và đưa vào những thông số cơ bản cho cổng kết nối như : Số hiệu cổng, baudrate, data bits, stop bits vào thì chương trình sẽ tự động tìm kiếm thiết bị. Một thiết bị Xbee mới sẽ phải nạp lại firmware. Chương trình XCTU đã hỗ trợ sẵn nên việc nạp rất dễ dàng.

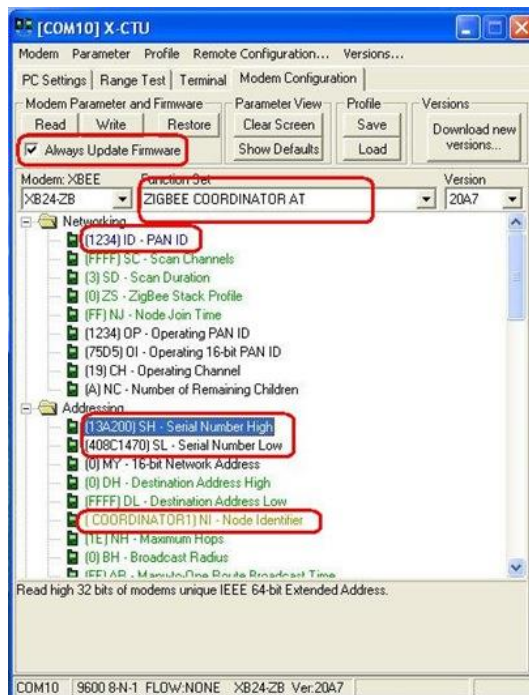


Hình 4.10 : Nhận diện Xbee

Thiết bị đã được cài đặt là Modem type = XB24-B và có phiên bản phần cứng là 20A7. Số Serial là 13A200408C1470. Ba số liệu trên cực kỳ quan trọng với Xbee. Với hai thông số trước xác định đời của Xbee và bình thường trong một mạng cảm biến có số lượng lớn Xbee thì việc cài đặt cùng firmware là cần thiết và hợp lý. Còn số Serial là đại diện cho Xbee đó, nhà sản xuất lấy đó làm địa chỉ cho Xbee luôn. Cho nên mỗi một Xbee đều có địa chỉ chính thức. Bình thường mỗi một gói tin trong Xbee được chuyển đi đều có địa chỉ đi và đến. Một mạng ZigBee thường có rất rất nhiều thiết bị nên việc lưu trữ địa chỉ này có thể không hợp lý. Cho nên nhà sản xuất cho phép gói tin được phát tán xung quanh theo cách BroadCast. Một gói tin được gửi không cần phải có đích đến, và những nút xung quanh có thể nhận được đầy đủ gói tin đó. Tùy thuộc vào từng trường hợp để xử lý gói tin. Một gói tin BroadCast thường được gửi với mục đích nào đấy, cho nên trong gói tin sẽ có những tham số cần thiết. Những nút

xung quanh có thể dựa theo tham số đó để giữ gói tin hoặc bỏ qua gói tin đó. Việc gửi gói tin Broadcast sẽ rất có ích trong nhiều trường hợp. Ở đây khi xây dựng hệ thống có áp dụng thuật toán LEACH thì gói tin broadcast sẽ được áp dụng trong việc xác định nút chủ nhóm (cluster head) hoặc truyền thông báo của nút SINK đến tất cả các nút trong mạng.

Đối với những Xbee đã được sử dụng thì chương trình sẽ liệt kê những số liệu mà người dùng trước đã thiết lập. Như đã nêu trên hệ thống chúng ta cần một nút chủ (Sink) và sáu nút lá.



Hình 4.11 : Cấu hình Xbee

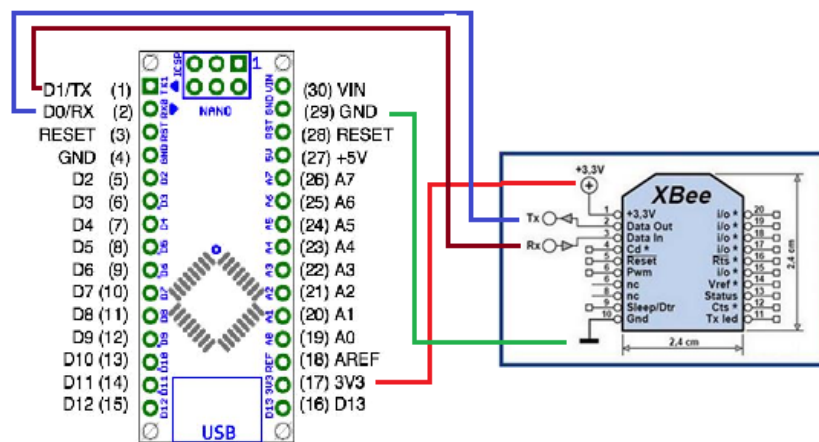
Hình trên hiển thị việc cài đặt một nút SINK ( Cũng có thể gọi là nút Coordinator). Ở đây cấu hình khung truyền AT. Để giúp việc định dạng gói tin được nhanh và tiện dụng hơn ta có thể dùng khung truyền API. Chương trình XCTU hỗ trợ việc cập nhật firmware cho mỗi Xbee khi cấu hình. Một Xbee thường có rất nhiều thông số, trong đề tài này có những thông số quan trọng như sau:

- PAN ID : là ID của mạng cảm biến, tất cả các nút trong mạng nếu muốn truyền thông tin được cho nhau thì đầu tiên phải cùng mạng. Cho nên khi cài đặt nút chủ ( SINK) chúng ta phải ghi nhớ PAN ID của nó.

- Scan Channels : là kênh truyền. Trên mỗi mạng đều có thể phân ra nhiều kênh truyền khác nhau.
- Node ID : Là tên của Xbee ta đang cấu hình
- SH, SL : Chính là địa chỉ của Xbee đang cấu hình
- DH, DL : Là địa chỉ đích mà Xbee cấu hình muốn gửi thông tin, việc gửi gói tin broadcast thì chỉ để mặc định ở hai thông số này.

Còn có nhiều thông số quan trọng sẽ được nói đến ở phần sau. Ở phần tùy chọn function set là lựa chọn chức năng của Xbee. Hệ thống chúng ta đang cần gồm có một nút chủ ( SINK) thì lựa chọn sẽ là Coordinator API và sáu nút trong hai nhóm chúng ta sẽ chọn ZigBee End Decive (ZED). Sau khi cài đặt chức năng và các thông số cần thiết xong hãy ấn nút write để phần mềm bắt đầu cấu hình cho Xbee theo mong muốn.

Tiếp theo cần phải kết nối Xbee và Arduino, sơ đồ mạch kết nối được thể hiện như hình sau:



Hình 4.12 : Sơ đồ mạch kết nối Xbee và Arduino Nano (<http://arduino.vn/>)



```

#include <XBee.h>
XBee xbee = XBee();
XBeeResponse response = XBeeResponse();
ZBRxResponse rx = ZBRxResponse();
ModemStatusResponse msr = ModemStatusResponse();
void setup() {
  Serial.begin(9600);
  xbee.begin(Serial);
}
void loop() {
  xbee.readPacket();
  if (xbee.getResponse().isAvailable()) {
    if (xbee.getResponse().getApiId() == ZB_RX_RESPONSE)
      xbee.getResponse().getZBRxResponse(rx);
      // Xử lý gói tin đến rx
    }
    } else if (xbee.getResponse().isError()) {
    }
  }
}

```

Hình 4.13 : Code cho nút SINK ( Coordinator )

Như chúng ta đã biết Arduino sử dụng ngôn ngữ C nên muốn thao tác với Xbee ban đầu phải có thư viện Xbee.h được cung cấp trên trang chủ.

```

XBee xbee = XBee();
XBeeResponse response = XBeeResponse();
ZBRxResponse rx = ZBRxResponse();
ModemStatusResponse msr = ModemStatusResponse();

```

Những câu lệnh này dùng để khai báo một thiết bị Xbee trong hệ thống.

```

void setup() {
  Serial.begin(9600);
  xbee.begin(Serial);
}

```

Hàm setup() sẽ được khởi chạy khi Arduino Nano được bật lên. Lúc này Xbee sẽ hoạt động với hai chân Rx,Tx của Arduino Nano. Serial.begin(9600) là khởi động cổng tuyến thông nối tiếp của Arduino Nano và máy tính với Data bits là 9600. Tiếp đến hàm loop() là một vòng lặp tuần hoàn của arduino nano. Trong suốt khoảng thời gian hoạt động của arduino nano nếu không gặp phải lỗi trong vòng lặp này thì vòng lặp sẽ được thực thi mãi mãi. Cho nên mỗi vòng lặp thì Xbee đều được kiểm tra xem có gói tin mới nào đến không. Mỗi vòng lặp được thực thi rất nhanh nên những gói tin được truyền liên tục từ những nút khác về gần như không thất thoát.

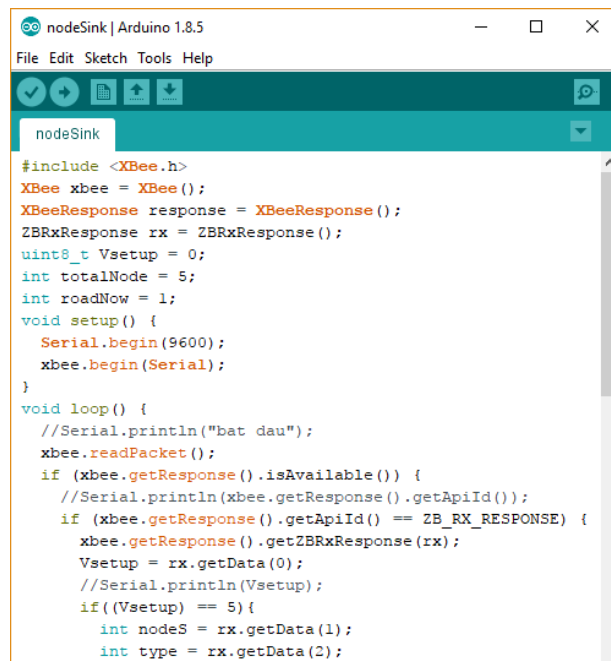
Tiếp đến sẽ nạp code cho các nút lá trong cụm đã được thiết lập trước. Cụm này gồm có bốn nút, mỗi nút sẽ đảm nhận gửi những gói tin về nút chủ (Coordinator). Trong gói tin ta có thể thiết lập để có thể phân biệt gói tin đến từ đâu và nội dung gói tin là gì.

Như vậy bước đầu tôi đã xây dựng được một mạng ZigBee đơn thuần sử dụng thiết bị truyền thông Xbee. Bước tiếp theo cũng là bước quan trọng nhất của đề tài. Tôi tiến hành phân tích giao thức LEACH và áp dụng thuật toán LEACH vào trong ZigBee. Hệ thống chúng ta gồm có một nút chủ (SINK) và sáu nút thành phần. Sáu nút thành phần sẽ xây dựng thành hai cụm, và sau một chu kỳ sẽ tiến hành lựa chọn chủ cụm (Cluster Head). Phần sau sẽ trình bày chi tiết về việc xây dựng hệ thống.

#### **4.3. Xây dựng nút SINK:**

Cũng giống mọi mạng cảm biến thì nút SINK nhằm mục đích thu thập thông tin của các nút thành phần trong mạng. Là trung tâm của một mạng nên nút SINK thường được thiết kế với bo mạch mạnh, có thể hoạt động thường xuyên, liên tục. Nguồn cung cấp cho nút SINK thường là nguồn điện cố định, trong nút SINK sẽ sử dụng một phương pháp nào đấy để vận chuyển thông tin thu thập đến với người dùng.

Như đã giới thiệu về giao thức LEACH ở trên, sau mỗi chu kỳ sẽ lựa chọn ra một nút cluster head. Việc chọn ra một nút chủ nhóm (Cluster Head) phải được thực thi khi chu kỳ mới bắt đầu và được tất cả các nhòm trong cụm tham gia đầy đủ. Như vậy việc định chu kỳ của hệ thống sẽ được nút SINK quyết định. Mỗi nút thành phần chỉ việc lắng nghe và khi nhận được tín hiệu từ nút SINK thì xác định một chu kỳ mới hình thành. Xây dựng nút SINK qua ngôn ngữ C được nạp vào Arduino Nano gắn trên nút SINK. Mỗi một vòng lặp (loop) trong nút SINK sẽ kiểm tra xem có dữ liệu được gửi về không và kiểm tra xem đã phải thông báo với tất cả nút thành phần là chu kỳ mới bắt đầu hay chưa.



```
#include <XBee.h>
XBee xbee = XBee();
XBeeResponse response = XBeeResponse();
ZBRxResponse rx = ZBRxResponse();
uint8_t Vsetup = 0;
int totalNode = 5;
int roadNow = 1;
void setup() {
  Serial.begin(9600);
  xbee.begin(Serial);
}
void loop() {
  //Serial.println("bat dau");
  xbee.readPacket();
  if (xbee.getResponse().isAvailable()) {
    //Serial.println(xbee.getResponse().getApiId());
    if (xbee.getResponse().getApiId() == ZB_RX_RESPONSE) {
      xbee.getResponse().getZBRxResponse(rx);
      Vsetup = rx.getData(0);
      //Serial.println(Vsetup);
      if (Vsetup == 5) {
        int nodeS = rx.getData(1);
        int type = rx.getData(2);
```

Hình 4.15 : Giao diện Arduino IDE

Dùng biến Vsetup để phân biệt gói tin được phát từ nút SINK và gói tin được phát từ các nút khác, biến info là nội dung gửi đi và broadcast là số lần gửi broadcast của nút SINK. Nút SINK thông báo chu kỳ bắt đầu. Vì mỗi chu kỳ tất cả các nút thành phần đều phải tham gia cho nên nút SINK sẽ phải gửi một gói tin quảng bá ( Broadcast ) đến tất cả gói tin trong mạng. Gói tin thông báo phải có tỉ lệ nhận được cao cho nên nút SINK phải gửi gói tin liên tục và nhiều lần trong một khoảng thời gian nhất định. Nút SINK phát gói tin broadcast đi bốn lần, mỗi lần sẽ tạm dừng trong 0,3 giây. Vậy sau bốn lần thời gian sẽ vượt qua ngưỡng 1 giây và vòng if sẽ không lặp lại một lần nữa trong chu kỳ này. Lúc này ta tăng biến roundNow lên một đơn vị. Cấu trúc gói tin gồm có Vsetup để đánh dấu gói tin broadcast từ nút SINK, totalNode là số lượng node trong hệ thống, roadNow là vòng hiện tại. Muốn phát gói tin dạng broadcast thì ta đem tổng số vào trong địa chỉ là 0x0000 và 0xffff.

Như vậy ta đã xây dựng được một nút SINK luôn lắng nghe thông tin từ các nút trong hệ thống mạng. Và đồng thời sau 20 phút sẽ phát đi gói tin broadcast để thông báo cho tất cả các nút thành viên là vòng tiếp theo sẽ bắt đầu.



#### **4.4. Xây dựng nút Cluster Head:**

Tiếp theo chúng ta sẽ xây dựng các nút thành phần. Nút thành phần có chức năng đảm nhận công việc chính là truyền thông tin trong mạng. Như giao thức LEACH đã nói trên, một chức năng nữa của nút thành phần là chuyển tiếp các gói tin trong cụm các nút đến cho nút SINK. Ở mạng Zigbee đơn thuần thì nút đảm nhận công việc chuyển tiếp gói tin chính là nút ZR ( Zigbee Router ). Như vậy chúng ta thấy khi áp dụng LEACH vào trong Zigbee thì sẽ không cần đến nút ZR nữa mà thay vào đó các nút thành phần sẽ thay phiên nhau làm công việc chuyển tiếp các gói tin trong mạng.

Câu lệnh `if(roundNow != roundOld)` dùng để kiểm tra vòng lặp mới. Khi có vòng lặp mới ta sẽ đặt hai biến `checkRoad` và `getCluster` về giá trị mặc định.

Chức năng chính của nút không phải ClusterHead là đóng gói dữ liệu thu thập được từ các cảm biến xung quanh và thêm vào một trường dữ liệu để xác định gói tin thứ bao nhiêu đã được gửi đi.

Biến `hieusuatX` dùng để lưu trữ hiệu suất truyền năng lượng từ nút đang xét đến nút gửi dữ liệu. Như đã trình bày ở phần trước nút thành phần sẽ chọn nút có giá trị hiệu suất tốt nhất thành nút Cluster Head quản lý mình.

#### **4.5. Lắp đặt chạy thử hệ thống:**

Như đã trình bày ở trên về các bước xây dựng nút SINK và nút thành phần. Chúng ta tiến hành lắp đặt Xbee vào các Arduino Nano vừa mới được nạp code. Phải phân biệt rõ đâu là Xbee dành cho nút SINK và Xbee dành cho nút thành phần. Sau đó chúng ta tiến hành cấp nguồn cho hệ thống bằng cách cắm cáp USB hoặc thiết kế bộ nguồn riêng cho từng nút. Khởi động nút SINK đầu tiên để Xbee ở nút SINK xây dựng mạng Zigbee. Như vậy khi các nút thành phần khởi động sẽ tự động tham gia ( `join` ) vào mạng Zigbee vừa rồi.

#### **4.6. Lắp đặt hệ thống đo mức tiêu thụ điện:**

Do chưa đủ điều kiện để xây dựng một hệ thống đo mức tiêu thụ điện năng riêng cho thiết bị nên tôi sử dụng pin sạc dự phòng có màn hình LCD theo dõi dung lượng pin.

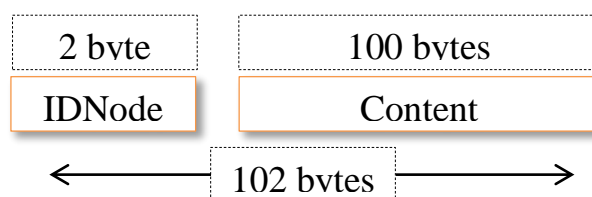


Hình 4.19: Pin sạc dự phòng

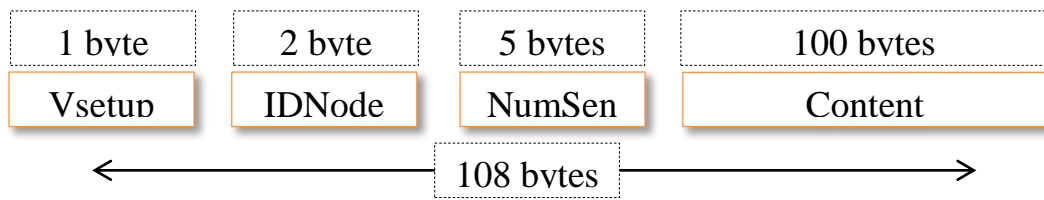
Màn hình LCD của pin sẽ hiển thị phần trăm năng lượng của pin, qua đó khi một thiết bị sử dụng pin một thời gian ta sẽ xác định được trong khoảng thời gian đó thiết bị tiêu thụ bao nhiêu pin. Pin của tôi có dung lượng 10000 mAh và được sạc đầy trước mỗi lần đo.

Nội dung của phần này là đo mức năng lượng của thiết bị và so sánh ưu điểm phương pháp mới tôi đề xuất. Cho nên tôi so sánh giữa mạng cảm biến không dây ZigBee thuần, giao thức LEACH và mạng cảm biến ZigBee được cải tiến theo phương pháp của tôi, cả ba mạng cảm biến đều sử dụng thiết bị truyền thông không dây Xbee.

Khi thực hiện việc đo đạc và so sánh tôi tập trung vào ba chỉ số chính: thời gian hoạt động, dung lượng thông tin, mức năng lượng tiêu thụ. Mạng cảm biến không dây trên thực tế thường được thiết lập hoạt động ở mức độ thấp và thời gian hoạt động của hệ thống dài. Cho nên muốn đánh giá được một mạng cảm biến không dây cụ thể phải cần có thời gian. Do thời gian không nhiều nên tôi tăng cường độ hoạt động và dung lượng thông tin truyền tải trong hệ thống nhằm rút ngắn thời gian thí nghiệm, đo đạc của tôi.



Hình 4.19. Gói tin trong ZigBee thuần



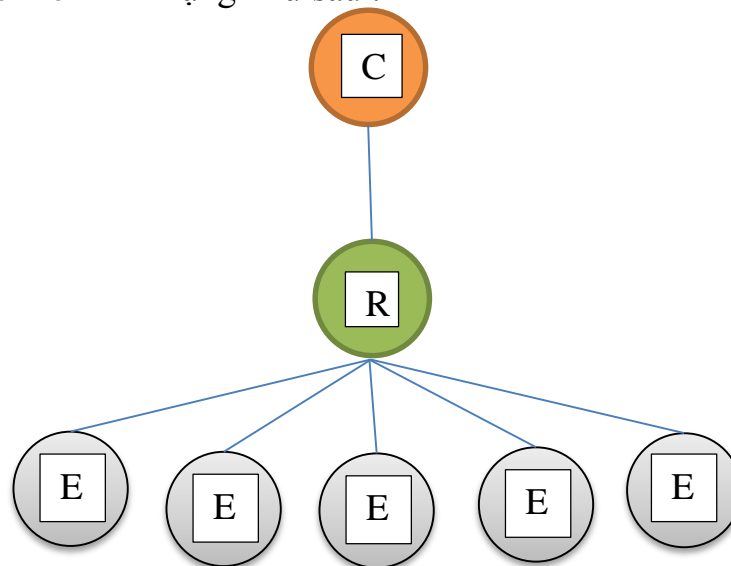
Hình 4.20. Gói tin trong phương pháp mới của tôi

Như đã giới thiệu ở phần trước, thiết kế cấu trúc gói tin của ZigBee thuần khác với cấu trúc gói tin của phương pháp mới cho nên để so sánh dung lượng thông tin truyền tải trong mạng tôi sẽ tính dung lượng của nội dung gói tin. Nội dung gói tin trong thí nghiệm của tôi là :

“anhsang:205;nhietdokhongkhi:25;nhietdogiathe:30;doamkhongkhi:70;no ngdoco2:4.84;nongdoph:6.54”

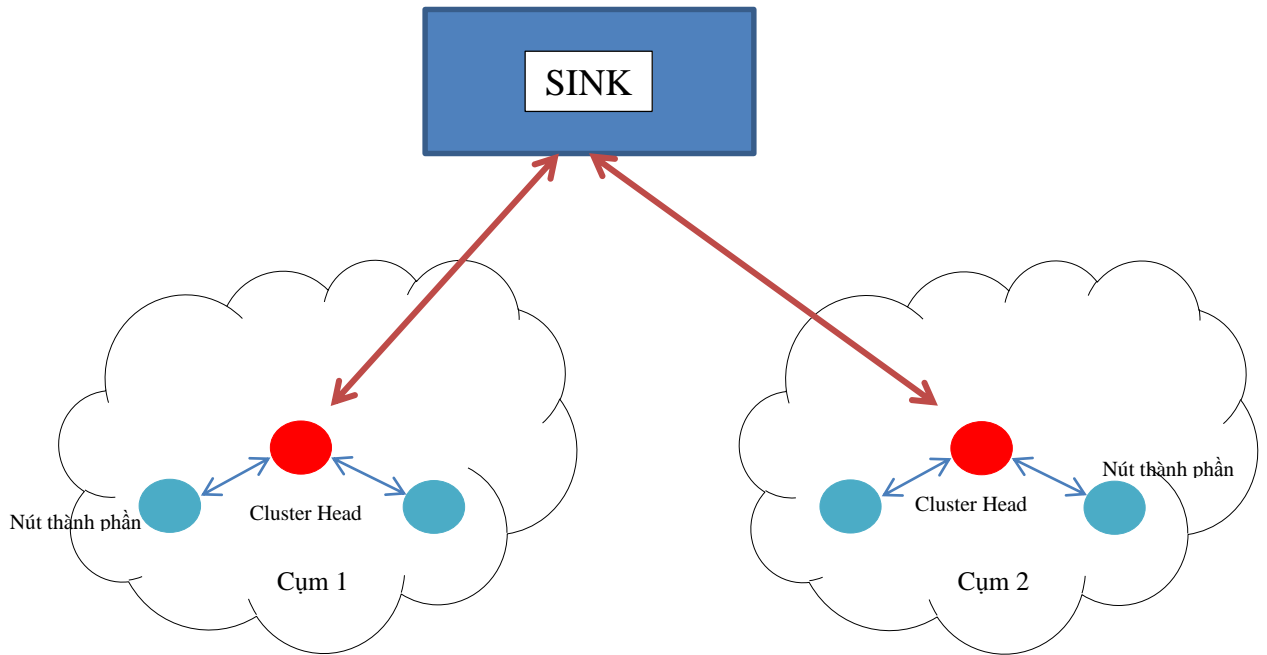
Tổng dung lượng nội dung gói tin là 108 Byte. Chu kỳ gửi gói tin của các nút thành phần là 2s.

Hệ thống của tôi gồm có 7 node (mỗi node gồm thiết bị truyền thông Xbee và bo mạch arduino), như vậy khi lắp đặt mạng truyền thông không dây ZigBee thuần ta có mô hình mạng như sau :



Hình 4.21. Mô hình 7 node ZigBee thuần

Mô hình gồm có một nút chủ C (codinator), 1 nút router R và 5 nút thành phần. Khi tính toán và đo lường ta sẽ đo cho hai thiết bị R và E như vậy sẽ đánh giá được phần nào mức độ tiêu thụ năng lượng của toàn mạng. Dưới đây là mô hình mạng cảm biến theo phương pháp tôi đặt ra với 7 node cơ bản.



Hình 4.22. Mô hình mạng của phương pháp mới với 7 node

Mô hình gồm 1 nút chủ (SINK) và 6 nút thành phần. Ở mô hình mạng phương pháp mới ta chỉ cần đo bất kỳ một nút thành phần nào trong mạng là có thể đánh giá mức tiêu thụ điện năng cho mạng.

Mô hình cho giao thức LEACH tôi cũng thiết kế tương tự với phương pháp mới của tôi.

Tôi đã thực nghiệm rất nhiều lần trên cả ba hệ thống và có những kết quả cụ thể như thông kê phía dưới:

- **Thời gian hệ thống hoạt động 1 tiếng:**

Với hệ thống ZigBee cơ bản:

Thiết bị sử dụng Pin	Tổng số gói tin gửi đi	Tổng số gói tin nhận được	Thời gian hệ thống hoạt động	Dung lượng pin còn lại(%)	Dung lượng Pin tiêu thụ (%)
Nút Router		7841	1 Giờ	91	9
Nút ZED	1800		1 Giờ	98	2

Hệ thống bao gồm 1 nút Coordinator, 1 nút router và 5 nút ZED như vậy có thể tính được các chỉ số sau:

Tổng điện năng tiêu thụ của hệ thống không tính nút Coordinator là :

$$9 + 2*5 = 19 \% (Pin sạc)$$

Tỷ lệ gói tin nhận được là :

$$((7841/5)*100)/1800 = 87\%$$

Với giao thức LEACH:

Thiết bị sử dụng Pin	Tổng số gói tin gửi đi	Tổng số gói tin nhận được	Thời gian hệ thống hoạt động	Chu kỳ chọn Cluster Head	Dung lượng pin còn lại(%)	Dung lượng Pin tiêu thụ (%)
Nút SINK		9381				
Nút thành phần	1800		1 Giờ	20 Phút	97	3

Hệ thống bao gồm một nút SINK và 6 nút thành phần như vậy có thể tính được những chỉ số sau:

Tổng điện năng tiêu thụ của hệ thống không tính nút SINK là :

$$3*6 = 18 \% (Pin sạc)$$

Tỷ lệ gói tin nhận được là :

$$((9381/6)*100)/1800 = 86.8\%$$

Với hệ thống ZigBee áp dụng phương pháp mới:

Thiết bị sử dụng Pin	Tổng số gói tin gửi đi	Tổng số gói tin nhận được	Thời gian hệ thống hoạt động	Chu kỳ chọn Cluster Head	Dung lượng pin còn lại(%)	Dung lượng Pin tiêu thụ (%)
Nút thành phần	1800	1593	1 Giờ	10 Phút	96	4
Nút thành phần	1800	1585	1 Giờ	20 Phút	97	3

Hệ thống bao gồm một nút SINK và 6 nút thành phần như vậy có thể tính được những chỉ số sau:

Tổng điện năng tiêu thụ của hệ thống không tính nút SINK là :

$$(4*6 + 3*6)/2 = 21 \% (Pin sạc)$$

Tỷ lệ gói tin nhận được là :

$$1590*100/1800 = 88.3\%$$

- **Thời gian hệ thống hoạt động 2 tiếng:**

Với hệ thống ZigBee cơ bản:

Thiết bị sử dụng Pin	Tổng số gói tin gửi đi	Tổng số gói tin nhận được	Thời gian hệ thống hoạt động	Dung lượng pin còn lại(%)	Dung lượng Pin tiêu thụ (%)
Nút Router		15640	2 Giờ	76	24
Nút ZED	3600		2 Giờ	95	5

*Hệ thống bao gồm 1 nút Coordinator, 1 nút router và 5 nút ZED như vậy có thể tính được các chỉ số sau:*

*Tổng điện năng tiêu thụ của hệ thống không tính nút Coordinator là :*

$$24 + 5*5 = 49 \% \text{ (Pin sạc)}$$

*Tỷ lệ gói tin nhận được là :*

$$((15640/5)*100)/3600 = 86.8\%$$

Với giao thức LEACH:

Thiết bị sử dụng Pin	Tổng số gói tin gửi đi	Tổng số gói tin nhận được	Thời gian hệ thống hoạt động	Chu kỳ chọn Cluster Head	Dung lượng pin còn lại(%)	Dung lượng Pin tiêu thụ (%)
Nút SINK		18760				
Nút thành phần	3600		2 Giờ	20 Phút	93	8

*Hệ thống bao gồm một nút SINK và 6 nút thành phần như vậy có thể tính được những chỉ số sau:*

*Tổng điện năng tiêu thụ của hệ thống không tính nút SINK là :*

$$8*6 = 48 \% \text{ (Pin sạc)}$$

*Tỷ lệ gói tin nhận được là :*

$$((18760/6)*100)/3600 = 86.8\%$$

Với hệ thống ZigBee áp dụng phương pháp mới:

Thiết bị sử dụng Pin	Tổng số gói tin gửi đi	Tổng số gói tin nhận được	Thời gian hệ thống hoạt động	Chu kỳ chọn Cluster Head	Dung lượng pin còn lại(%)	Dung lượng Pin tiêu thụ (%)
Nút thành phần	3600	3241	2 Giờ	10 Phút	91	9
Nút thành phần	3600	3199	2 Giờ	20 Phút	92	8

Hệ thống bao gồm một nút SINK và 6 nút thành phần như vậy có thể tính được những chỉ số sau:

Tổng điện năng tiêu thụ của hệ thống không tính nút SINK là :

$$(9*6 + 8*6)/2 = 51 \% \text{ (Pin sạc)}$$

Tỷ lệ gói tin nhận được là :

$$3199*100/3600 = 88.8\%$$

- **Thời gian hệ thống hoạt động 4 tiếng:**

Với hệ thống ZigBee cơ bản:

Thiết bị sử dụng Pin	Tổng số gói tin gửi đi	Tổng số gói tin nhận được	Thời gian hệ thống hoạt động	Dung lượng pin còn lại(%)	Dung lượng Pin tiêu thụ (%)
Nút Router		30931	4 Giờ	53	47
Nút ZED	7200		4 Giờ	89	11

Hệ thống bao gồm 1 nút Coordinator, 1 nút router và 5 nút ZED như vậy có thể tính được các chỉ số sau:

Tổng điện năng tiêu thụ của hệ thống không tính nút Coordinator là :

$$47 + 11*5 = 102 \% \text{ (Pin sạc)}$$

Tỷ lệ gói tin nhận được là :

$$((30931/5)*100)/7200 = 85.9\%$$

Với giao thức LEACH:

Thiết bị sử dụng Pin	Tổng số gói tin gửi đi	Tổng số gói tin nhận được	Thời gian hệ thống hoạt động	Chu kỳ chọn Cluster Head	Dung lượng pin còn lại(%)	Dung lượng Pin tiêu thụ (%)
Nút SINK		37149				
Nút thành phần	7200		4 Giờ	20 Phút	83	17

*Hệ thống bao gồm một nút SINK và 6 nút thành phần như vậy có thể tính được những chỉ số sau:*

*Tổng điện năng tiêu thụ của hệ thống không tính nút SINK là :*

$$17*6 = 102 \% \text{ (Pin sạc)}$$

*Tỷ lệ gói tin nhận được là :*

$$((37149/6)*100)/7200 = 86\%$$

Với hệ thống ZigBee áp dụng phương pháp mới:

Thiết bị sử dụng Pin	Tổng số gói tin gửi đi	Tổng số gói tin nhận được	Thời gian hệ thống hoạt động	Chu kỳ chọn Cluster Head	Dung lượng pin còn lại(%)	Dung lượng Pin tiêu thụ (%)
Nút thành phần	7200	6539	4 Giờ	10 Phút	82	18
Nút thành phần	7200	6544	4 Giờ	20 Phút	83	17

*Hệ thống bao gồm một nút SINK và 6 nút thành phần như vậy có thể tính được những chỉ số sau:*

*Tổng điện năng tiêu thụ của hệ thống không tính nút SINK là :*

$$(18*6 + 17*6)/2 = 105 \% \text{ (Pin sạc)}$$

*Tỷ lệ gói tin nhận được là :*

$$6502*100/7200 = 90.3\%$$



- **Thời gian hệ thống hoạt động 7 tiếng:**

Với hệ thống ZigBee cơ bản:

Thiết bị sử dụng Pin	Tổng số gói tin gửi đi	Tổng số gói tin nhận được	Thời gian hệ thống hoạt động	Dung lượng pin còn lại(%)	Dung lượng Pin tiêu thụ (%)
Nút Router		52359	7 Giờ	28	72
Nút ZED	12600		7 Giờ	79	21

*Hệ thống bao gồm 1 nút Coordinator, 1 nút router và 5 nút ZED như vậy có thể tính được các chỉ số sau:*

*Tổng điện năng tiêu thụ của hệ thống không tính nút Coordinator là :*

$$72 + 21 * 5 = 177 \% \text{ (Pin sạc)}$$

*Tỷ lệ gói tin nhận được là :*

$$((52359/5)*100)/12600 = 83.1\%$$

Với giao thức LEACH:

Thiết bị sử dụng Pin	Tổng số gói tin gửi đi	Tổng số gói tin nhận được	Thời gian hệ thống hoạt động	Chu kỳ chọn Cluster Head	Dung lượng pin còn lại(%)	Dung lượng Pin tiêu thụ (%)
Nút SINK		62975				
Nút thành phần	12600		7 Giờ	20 Phút	71	29

*Hệ thống bao gồm một nút SINK và 6 nút thành phần như vậy có thể tính được những chỉ số sau:*

*Tổng điện năng tiêu thụ của hệ thống không tính nút SINK là :*

$$29 * 6 = 174 \% \text{ (Pin sạc)}$$

*Tỷ lệ gói tin nhận được là :*

$$((62975/6)*100)/12600 = 83.3\%$$

Với hệ thống ZigBee áp dụng phương pháp mới:

Thiết bị sử dụng Pin	Tổng số gói tin gửi đi	Tổng số gói tin nhận được	Thời gian hệ thống hoạt động	Chu kỳ chọn Cluster Head	Dung lượng pin còn lại(%)	Dung lượng Pin tiêu thụ (%)
Nút thành phần	12600	11619	7 Giờ	10 Phút	70	30
Nút thành phần	12600	11621 kB	7 Giờ	20 Phút	72	28

Hệ thống bao gồm một nút SINK và 6 nút thành phần như vậy có thể tính được những chỉ số sau:

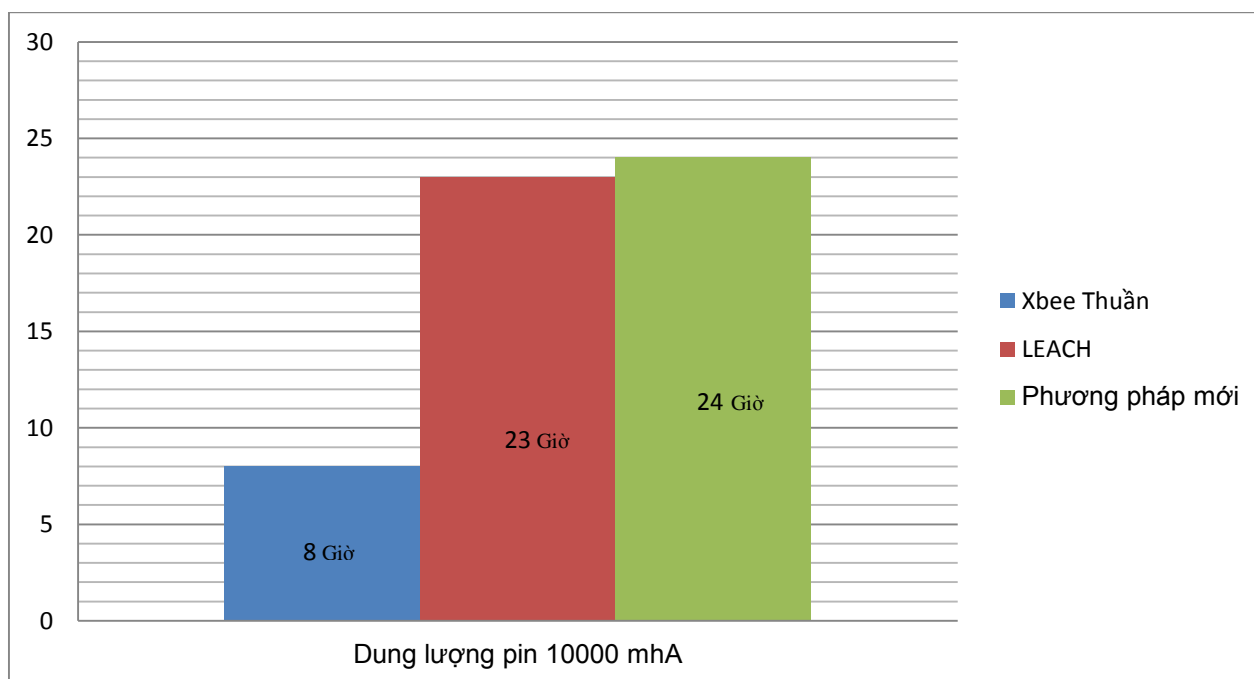
Tổng điện năng tiêu thụ của hệ thống không tính nút SINK là :

$$(30*6 + 28*6)/2 = 174 \% \text{ (Pin sạc)}$$

Tỷ lệ gói tin nhận được là :

$$11621*100/12600= 92.2\%$$

Với dung lượng pin 10000 mAh cho mỗi nút thành phần ta có thể ước tính thời gian sống của mạng như biểu đồ sau:

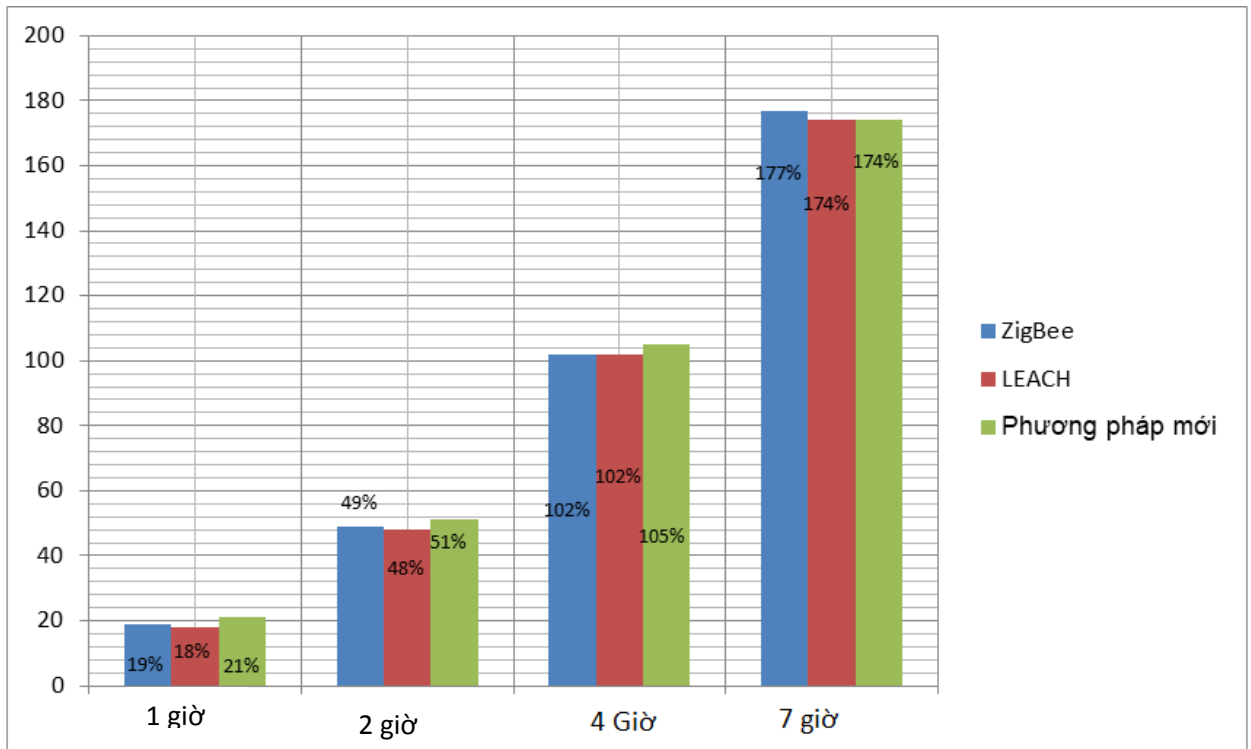


Hình 4.23. Biểu đồ so sánh thời gian sống của mạng

Như thực nghiệm ở trên nút Router của mạng ZigBee thuần hoạt động trong 7 giờ tiêu tốn hết 72% pin sạc vậy ước tính 8 giờ nút Router sẽ ngừng hoạt

động. Tương tự với các nút thành phần của LEACH và ZigBee áp dụng phương pháp mới thời gian sống của nút thành phần là 23 và 24 giờ.

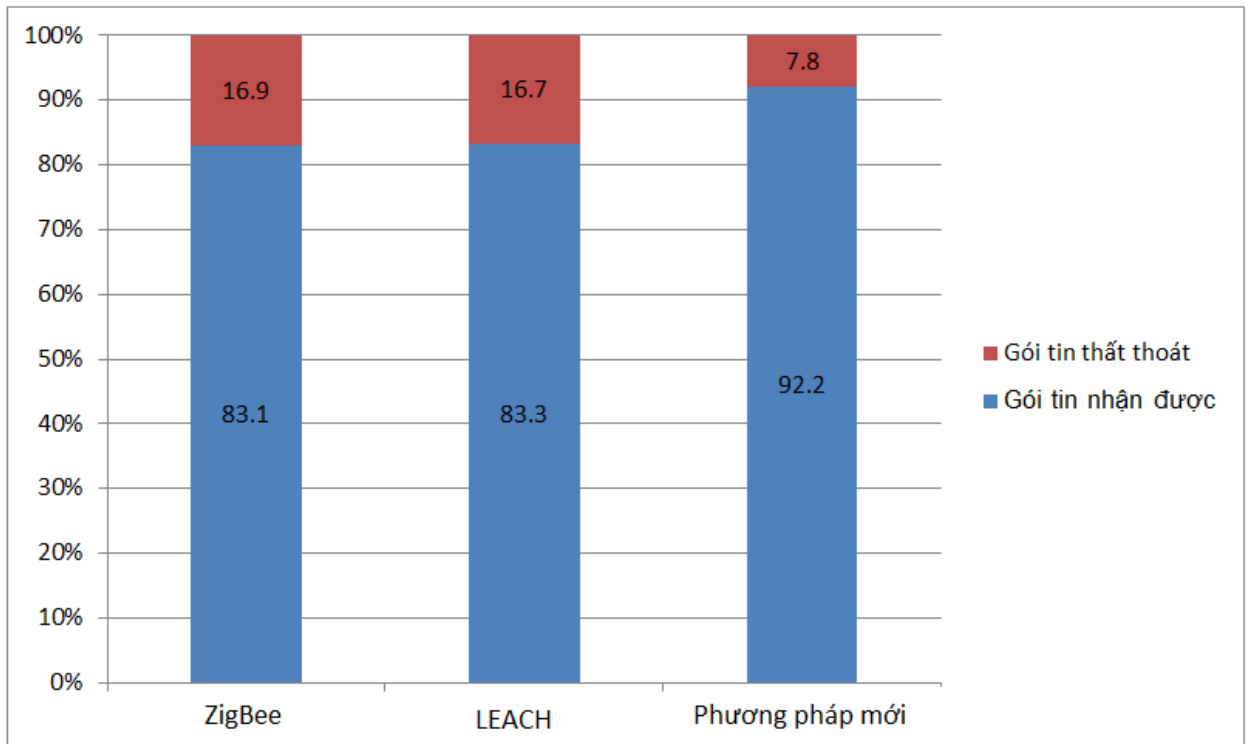
Tổng hợp những kết quả trên ta có biểu đồ so sánh tổng mức tiêu thụ điện năng:



Hình 4.24. Biểu đồ so sánh mức tiêu thụ điện năng

Giá trị tổng mức tiêu thụ điện năng được tổng hợp từ các thực nghiệm cụ thể đã thực hiện. Tổng mức tiêu thụ điện năng của phương pháp mới được tối ưu dần theo thời gian. Phương pháp mới cần một thời gian cài đặt và thu thập dữ liệu về hiệu suất truyền dữ liệu nên trong khoảng thời gian đầu tiêu thụ nhiều điện năng hơn. Nhưng khi bước vào giai đoạn ổn định thì phương pháp mới sẽ tiêu thụ ít năng lượng hơn. Nếu hệ thống hoạt động thời gian dài hơn thì phương pháp mới sẽ chứng minh được hiệu quả của mình.

Biểu đồ dưới đây thể hiện việc mất mát gói tin của ba hệ thống:



*Hình 4.25. Biểu đồ so sánh tỷ lệ mất mát gói tin trong hệ thống*

Biểu đồ trên thể hiện độ mất mát gói tin của ba hệ thống khi hoạt động trong 7 giờ. Với điều kiện địa lý giống nhau khi thực nghiệm trên ba hệ thống thì phương pháp mới có tỷ lệ mất mát gói tin thấp hơn hẳn.

## Chương V. KẾT LUẬN

Khoa học và công nghệ đã mang lại cho nhân loại những bước tiến vượt bậc về công nghệ. Mỗi một công nghệ ra đời đều được người dùng cảm nhận và kiểm chứng khắc khe. Công nghệ nào tiếp cận được với người dùng nhiều nhất chứng tỏ ưu điểm của công nghệ đó với các công nghệ khác cùng lĩnh vực. Mạng cảm biến không dây đã ra đời là sự kết hợp thành công của một loạt những thành tựu khoa học về công nghệ mạng máy tính. Hiện nay mạng cảm biến không dây đã đi vào nhiều lĩnh vực trong cuộc sống và nghiên cứu nhằm phục vụ con người. Với rất nhiều ưu điểm mang lại nên mạng cảm biến không dây đang khẳng định vị thế của mình trong công nghệ mạng truyền thông.

Để mang lại lợi ích tối ưu cho người sử dụng thì tốt nhất là tận dụng các điểm mạnh riêng biệt của mạng cảm ứng, đó là các sensor giá thành thấp, tiêu thụ ít năng lượng và có thể thực hiện đa chức năng. Tuy nhiên, WSN vẫn còn rất nhiều vấn đề cần hoàn thiện đặc biệt là vấn đề năng lượng và duy trì nguồn năng lượng cho các nút cảm biến. Đã rất nhiều nhà nghiên cứu đem ra rất nhiều giải pháp và hệ thống với mong muốn khắc phục dần điểm yếu của mạng cảm biến không dây.

Sau khi nghiên cứu và xây dựng được một mạng cảm biến không dây sử dụng phương thức LEACH và ZigBee tôi đã rút ra được những ưu điểm và những vấn đề gặp phải như sau:

- Ưu điểm :
  - Thời gian hoạt động của mạng được kéo dài
  - Tiết kiệm năng lượng
  - Tiết kiệm tài nguyên trong mạng
  - Hiệu suất truyền tin tốt hơn
- Vấn đề gặp phải :
  - Theo dõi, đối sánh cần thực hiện trong thời gian dài mới đảm bảo độ chính xác cao
  - Lượng công việc của khối điều khiển nhiều hơn

- Số lượng và chất lượng thiết bị còn bị hạn chế

Trong thời gian nghiên cứu và học tập tôi đã cố gắng khắc phục những vấn đề gặp phải đến mức thấp nhất. Khi hoàn thành nghiên cứu tôi đã thu được những kiến thức và kinh nghiệm mà đề tài mang lại. Tôi nhận thấy những ưu điểm trên khá tốt để cải thiện mạng cảm biến không dây ZigBee. Tôi hi vọng nghiên cứu này đóng góp một phần nào đó vào trong công cuộc phát triển mạng cảm biến không dây trên thế giới.

## TÀI LIỆU THAM KHẢO

### Tiếng Việt

- [1] [www.arduino.vn](http://www.arduino.vn) Website giới thiệu và hướng dẫn sử dụng Arduino.
- [2] <http://automation.net.vn>

### Tiếng Anh

- [3] Robert Faludi (2011), “*Building Wireless Sensor Networks: with ZigBee, XBee, Arduino, and Processing*”, O’Reilly Media-Inc.-1005 Gravenstein HighWay North – Sebastopol – CA 95472.
- [4] Vongsagon Boonsawat, Jurarat Ekchamanonta, Kulwadee Bumrunghet, and Somsak Kittipiyakul (June. 2012), “*XBee Wireless Sensor Networks for Temperature Monitoring*”, Thammasat University, Pathum-Thani, Thailand 12000
- [5] Nejla Rouissi, Hamza Gharsellaoui (2017), “*Improved Hybrid LEACH Based Approach for Preserving Secured Integrity in Wireless Sensor Networks*”, Carthage University, Tunisia
- [6] Geetha. V , Pranesh.V. Kallapur , Sushma Tellajeera (2012), “*Clustering in Wireless Sensor Networks: Performance Comparison of LEACH & LEACH-C Protocols Using NS2*”, Department of Information Technology, NITK, Surathkal, Karnataka, India
- [7] I.F. Akyildiz, W. Su\*, Y. Sankarasubramaniam, E. Cayirci (2011), “*Wireless sensor networks: a survey*”, Georgia Institute of Technology, Atlanta, GA 30332, USA
- [8] Yawei Zhao, Guohua Zhang, Zhongwu Xia, and Xinhua Li (2011), “*Improved ZigBee Network Routing Algorithm Based on LEACH*”, School of Information Science & Engineering, Shenyang Ligong University, 110159 Shenyang, China