



## BÀI GIẢNG MÔN

# ***Internet và giao thức*** ***(Internet and Protocols)***

**Giảng viên:**

**Ths. Hoàng Thị Thu**

**Điện thoại/E-mail:**

**0326189970**

**thuht@ptit.edu.vn**

**Bộ môn:**

**Mạng viễn thông - Khoa Viễn thông 1**

**Học kỳ/Năm biên soạn: I/ 2022-2023**

# Nội dung học phần Internet và giao thức (45 tiết=3tc, Lớp chính quy)

- **Lý thuyết:** 32 tiết
  - C1- Tổng quan về Internet
  - C2- Các ứng dụng và giao thức mạng
  - C3- Các giao thức ứng dụng đa phương tiện
  - ➡ ■ **C4- Giao thức ứng dụng Internet vạn vật**
    - 2 tiết kiểm tra
    - 2 tiết ôn tập
- **Bài tập:** 8 tiết – làm nhóm.
- **Thực hành:** 4 tiết – làm nhóm
- **Thi cuối kỳ:** Thi viết
- **Giờ tự học:** 1



## Chương 4: Giao thức ứng dụng Internet vạn vật

- Giới thiệu chung về IoT
- Kiến trúc và các ứng dụng IoT
- Các giải pháp công nghệ mạng IoT
- Các giao thức Request/Response
- Các giao thức Publish/Subscriber
- Kết luận chương

## 4.1. Giới thiệu chung về IoT

### IoT là gì?

IoT bao gồm một sự vật (Thing), thêm trí thông minh tính toán (Intelligence) để cải thiện chức năng của nó, sau đó thêm kết nối mạng (Internet) để tăng cường hơn nữa chức năng của nó.



*Hình 4.1: Mô hình IoT*



## 4.1. Giới thiệu chung về IoT

### Đặc điểm của IoT

- **Tính thông minh:** tự liên lạc với nhau để trao đổi thông tin và dữ liệu. Việc tích hợp ứng dụng IOT và hợp trí thông minh có thể giúp các thiết bị, máy móc và phần mềm thu thập được phân tích các dấu vết điện tử.
- **Kiến trúc dựa vào sự kiện:** IOT sẽ có thể phản hồi dựa vào các sự kiện được diễn ra trong lúc chúng hoạt động theo thời gian thực.
- **IOT là hệ thống phức tạp:** các liên kết giữa các thiết bị, máy móc và dịch vụ với nhau, ngoài ra IOT còn có khả năng thêm vào các nhân tố mới khác.



## 4.1. Giới thiệu chung về IoT

### Đặc điểm của IoT

- **Kích thước:** IOT là mạng lưới có khả năng chứa đến 50 đến 100 nghìn tỉ các đối tượng kết nối.
- **Không gian và thời gian:** IOT thu thập được nhiều dữ liệu với nhau, và các dữ liệu thừa về địa điểm, việc xử lý dữ liệu đó không hiệu quả => thách thức IoT.
- **Tính không đồng nhất:** Vì các thiết bị trong IoT có phần cứng khác nhau cũng như network khác nhau.
- **Tính kết nối liên thông (interconnectivity):** bất cứ một điều gì, vật gì hay máy móc gì cũng có thể được kết nối với nhau.



## 4.1. Giới thiệu chung về IoT

### Tiềm năng của IoT

- **Kết nối không giới hạn:** qua thiết bị thông minh
- **Phạm vi ứng dụng** trong các lĩnh vực như trí thông minh nhân tạo, robot, mạng Internet, phương tiện độc lập, điện toán đám mây, tích hợp thông tin, phân tích số liệu tự động,...
- **Thiết bị vận chuyển thông minh:** nhà máy và cánh đồng thông minh, những mạng lưới năng lượng và tiện ích (cung cấp nước, gas) thông minh



## 4.1. Giới thiệu chung về IoT

### Thách thức của IoT

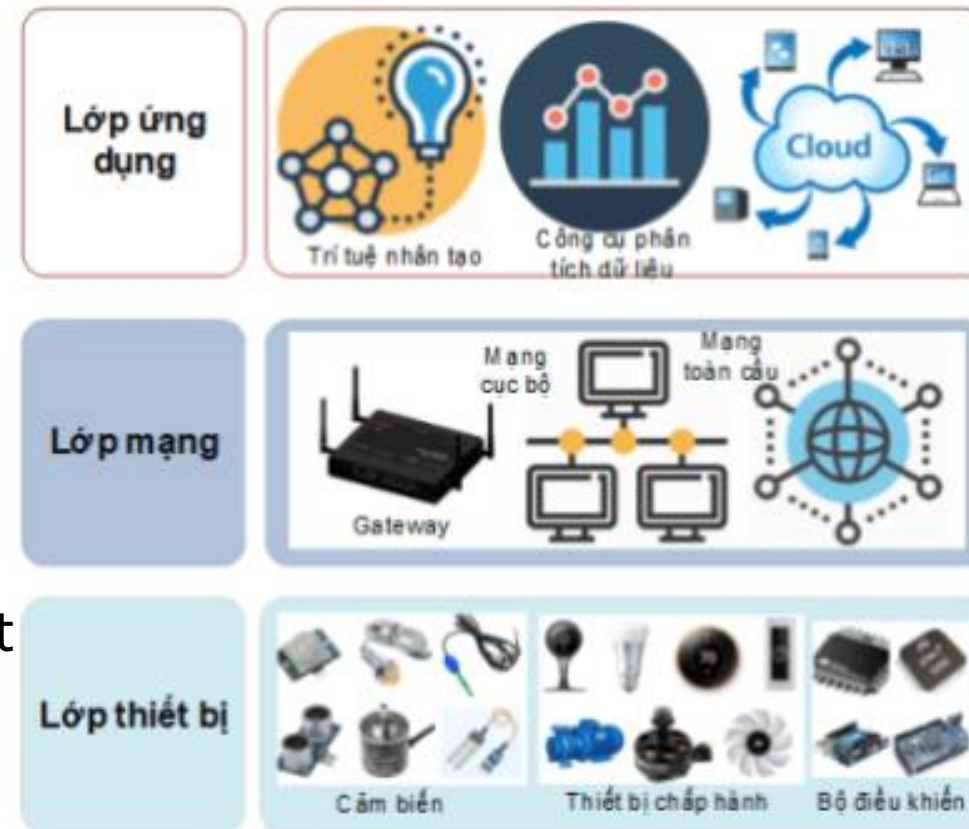
- **Bảo mật:** IoT là trung tâm dữ liệu mà tất cả các thiết bị/hệ thống kết nối hoạt động dựa trên dữ liệu có sẵn => luồng dữ liệu giữa các thiết bị, luôn có khả năng dữ liệu có thể bị truy cập hoặc đọc trong quá trình truyền.
- **Tính tương thích:** công nghệ trên mạng có thể được xác định chính xác là dành cho IoT gồm: Bluetooth, Bluetooth LE, ZigBee, RFID, Wi-Fi, Z-Wave, 6LowPAN, NFC, Sigfox => tiêu chuẩn kỹ thuật khác nhau => tương tác khó khăn.



## 4.2. Kiến trúc và các ứng dụng IoT

### Kiến trúc hệ thống IoT

- **Lớp thiết bị:** cảm biến, thiết bị chấp hành và các bộ điều khiển như vi xử lý/vi điều khiển, PLC, FPGA đến các máy tính nhúng.
- **Lớp mạng:** xác định các giao thức truyền thông được sử dụng cho việc kết nối mạng và thực hiện điện toán biên.
- **Lớp ứng dụng:** thu nhận dữ liệu từ lớp mạng, lưu trữ, xử lý dữ liệu và ra quyết định dựa trên các thuật toán AI/ML hoặc các công cụ phân tích dữ liệu hiện đại.



Hình 4.2: Mô hình kiến trúc IoT



## 4.2. Kiến trúc và các ứng dụng IoT

**Một hệ thống IoT sẽ bao gồm 4 thành phần:**

1. Thiết bị hay còn được gọi là Things;
2. Trạm kết nối hay là cổng kết nối (Gateways);
3. Hạ tầng mạng hay là các điện toán đám mây (Network and Cloud);
4. Bộ phân tích và xử lý dữ liệu (Services-creation and Solution Layers).

## 4.2. Kiến trúc và các ứng dụng IoT

### Luồng quy trình dữ liệu trong IoT

Kết nối các cảm biến, bộ truyền động với vạn vật (things)



**Internet Gateways:** tổng hợp và chuyển đổi thành dạng kỹ thuật số để xử lý, kết nối với phần còn lại của hệ thống



**Hệ thống biên (edge IT system):** giảm tải cho cơ sở hạ tầng IT cốt lõi



**Data center/Cloud platform:** quản lý, phân tích và lưu trữ dữ liệu an toàn



## 4.2. Kiến trúc và các ứng dụng IoT

### Các ứng dụng IoT

#### **(1) Ứng dụng trong việc xây dựng nhà thông minh**

Nhà thông minh (home automation, domotics, smart home hoặc Intellihome) là kiểu nhà được lắp đặt các thiết bị điện, điện tử được điều khiển/tự động hoá/bán tự động, thay thế con người trong thực hiện một hoặc một số thao tác quản lý, điều khiển.

- Điều khiển chiếu sáng (on/off, dimmer, scene, timer, logic,...)
- Điều khiển màn hình, rèm, cửa, cổng
- Hệ thống an ninh, báo động, báo cháy
- Điều khiển điều hòa, máy lạnh
- Hệ thống âm thanh đa vùng
- Camera, chuông hình
- Hệ thống bảo vệ nguồn điện



## 4.2. Kiến trúc và các ứng dụng IoT

### Các ứng dụng IoT

#### ***Ứng dụng trong thành phố thông minh/Đô thị thông minh***

Mô hình thành phố ứng dụng công nghệ thông tin, trí tuệ nhân tạo để quản lý, nâng cao tiêu chuẩn cuộc sống đô thị, cải thiện chất lượng phục vụ của chính quyền thành phố và sử dụng hiệu quả các nguồn năng lượng, tài nguyên thiên.

- Cung cấp nước hiệu quả hơn
- Giải quyết vấn nạn ùn tắc giao thông
- Nâng cao độ tin cậy, chất lượng của giao thông công cộng.



## 4.2. Kiến trúc và các ứng dụng IoT

### Các ứng dụng IoT

#### ***Ứng dụng trong sản xuất công nghiệp***

- IIoT (Industrial Internet of Things): sử dụng các thiết bị có kết nối mạng và trí tuệ nhân tạo trong nhà máy.
- IIoT cải thiện khả năng kết nối, tiết kiệm thời gian, chi phí cho các tổ chức.
- IIoT cho phép các tổ chức truy cập nguồn dữ liệu khổng lồ, kết nối mọi người, dữ liệu, quy trình từ các nhà máy tới người quản lý.



## 4.2. Kiến trúc và các ứng dụng IoT

### Các ứng dụng IoT

#### ***Ứng dụng trong chăm sóc sức khỏe và y tế***

- Thực hiện các hành động như cung cấp cảnh báo hoặc gửi thông báo, khi giá trị đo đạt đến một mức nhất định.
- Đưa ra khuyến nghị cho bệnh nhân hoặc chuyên gia chăm sóc sức khỏe dựa trên phân tích dữ liệu.
- Khả năng cung cấp thuốc dựa trên dữ liệu bệnh nhân được theo dõi và hướng dẫn y tế.



## 4.2. Kiến trúc và các ứng dụng IoT

### Các ứng dụng IoT

#### *Ứng dụng trong nông nghiệp*

- ***Giám sát điều kiện khí hậu:*** lập bản đồ các điều kiện khí hậu, chọn các loại cây trồng phù hợp.
- ***Tự động hóa nhà kính:*** **Hệ thống điều khiển tưới** gồm đầu tưới nhỏ giọt/đầu tưới phun sương/mưa, bộ điều khiển tưới. **Hệ thống điều khiển khí hậu** gồm cảm biến nhiệt độ, độ ẩm bên trong và bên ngoài nhà kính, quạt thông gió, đèn chiếu sáng.
- ***Quản lý cây trồng:*** theo dõi sự tăng trưởng của cây trồng
- ***Giám sát và quản lý gia súc:*** theo dõi sức khỏe, hiệu suất
- ***Hệ thống quản lý trang trại đầu cuối:*** giám sát trang trại từ xa và cho phép người dùng hợp lý hóa các hoạt động kinh doanh.

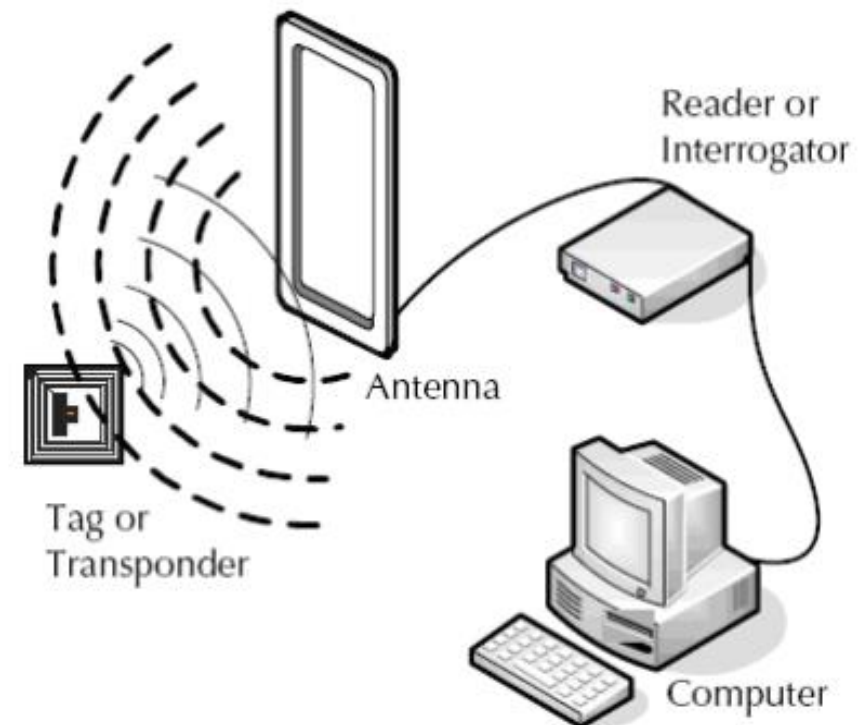


## 4.3. Các giải pháp công nghệ mạng IoT

**RFID:** Radio Frequency Identification (Nhận dạng bằng sóng vô tuyến)

Công nghệ này sử dụng sóng vô tuyến => tự động nhận diện và theo dõi vật thể.

Vật thể này có thể là bất cứ thứ gì, chẳng hạn như một cuốn sách trong thư viện, một món đồ trong siêu thị, hàng để trong kho, thậm chí là một chiếc ô tô hay hay động vật.



Hình 4.3: Ví dụ ứng hệ thống RFID



## 4.3. Các giải pháp công nghệ mạng IoT

**RFID:** Radio Frequency Identification (Nhận dạng bằng sóng vô tuyến)

- **Tag:** gắn chip + antenna, được lập trình điện tử với thông tin duy nhất => lưu trữ và truyền dữ liệu đến một đầu đọc trong một môi trường tiếp xúc bằng sóng vô tuyến.
- **Reader:** thiết bị kết nối không dây với tag để dễ dàng nhận dạng đối tượng được gắn tag.
- **Antenna thu, phát sóng vô tuyến:** thiết bị liên kết giữa thẻ và thiết bị đọc.
- **Máy chủ và hệ thống phần mềm:** RFID có thể hoạt động độc lập không có thành phần này.
- **Cơ sở tầng truyền thông:** giao tiếp với hệ thống



## 4.3. Các giải pháp công nghệ mạng IoT

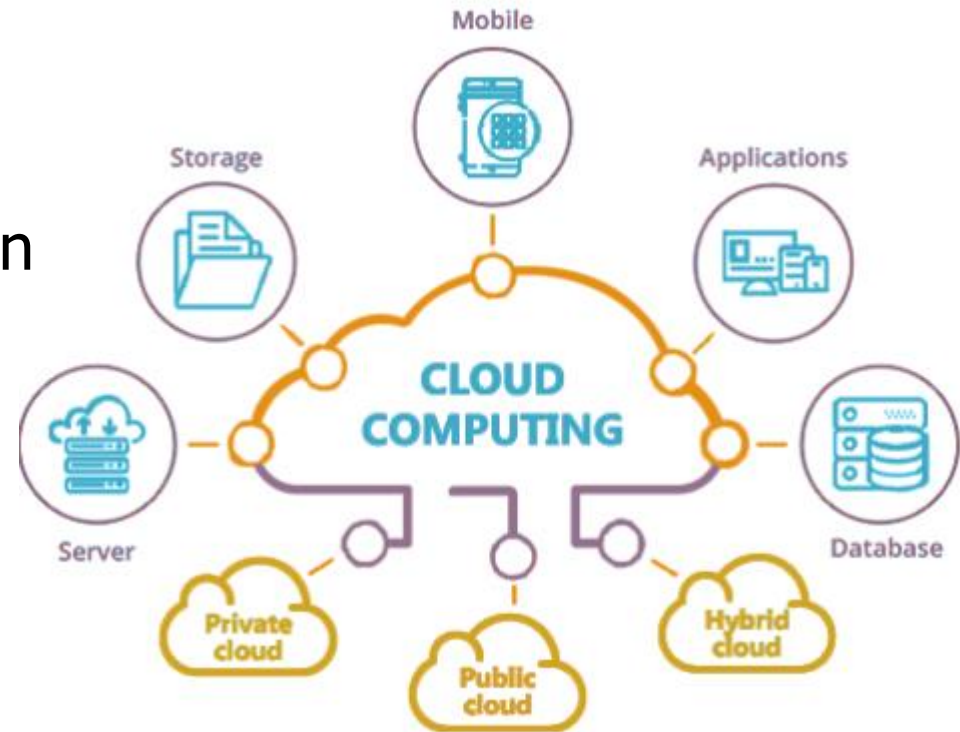
**RFID:** Radio Frequency Identification (Nhận dạng bằng sóng vô tuyến)

- Trong IoT, RFID được sử dụng cùng với camera, GPS, cảm biến thông minh để định vị và xác định các đối tượng.
- Có 3 loại tag RFID:
  - Active tag (chủ động):** có nguồn riêng bên trong, truyền dữ liệu bằng nguồn đó
  - Passive tag (thụ động):** không có nguồn bên trong, sử dụng nguồn nhận được từ reader để hoạt động và truyền dữ liệu được lưu trữ trong nó cho reader
  - Semi-passive tag (bán thụ động):** có nguồn riêng bên trong, sử dụng nguồn từ reader để truyền dữ liệu, truyền được khoảng cách xa hơn so với tag thụ động.

## 4.3. Các giải pháp công nghệ mạng IoT

### ***Cloud Computing*** (Điện toán đám mây)

Mô hình cung cấp các tài nguyên máy tính cho người dùng thông qua Internet. Nguồn tài nguyên này bao gồm rất nhiều thứ liên quan đến điện toán và máy tính. Ví dụ như: phần mềm, dịch vụ, phần cứng,... và sẽ nằm tại các máy chủ ảo (đám mây) trên mạng.



Hình 4.4: Điện toán đám mây



## 4.3. Các giải pháp công nghệ mạng IoT

### *Cloud Computing* (Điện toán đám mây)

#### Mô hình triển khai điện toán đám mây

- **Public Cloud:** tất cả người dùng sẽ dùng chung tài nguyên.
- **Private Cloud:** cung cấp cho các doanh nghiệp để đảm bảo an toàn dữ liệu.
- **Đám mây lai (Hybrid Cloud)** là sự kết hợp giữa đám mây công cộng và đám mây riêng.
- **Đám mây cộng đồng (Community Cloud)** được xây dựng nhằm mục đích chia sẻ hạ tầng, dữ liệu cho nhiều tổ chức, người dùng khác nhau.



## 4.3. Các giải pháp công nghệ mạng IoT

### *Cloud Computing* (Điện toán đám mây)

#### Mô hình cung cấp điện toán đám mây

- ***Iaas (Infrastructure as a service)***: pay-per-use (trả tiền cho những gì sử dụng). Chi phí sử dụng dịch vụ này được tính dựa trên chức năng và lượng tài nguyên mà khách hàng dung.
- ***Paas (Platform as a service)***: mô hình dịch vụ giúp các developer có thể phát triển, cho phép triển khai các ứng dụng, website trên đám mây.
- ***Saas (Software as a service)***: cho phép người dùng sử dụng được các ứng dụng trên nền tảng đám mây thông qua internet.



## 4.3. Các giải pháp công nghệ mạng IoT

### ***Cloud Computing*** (Điện toán đám mây)

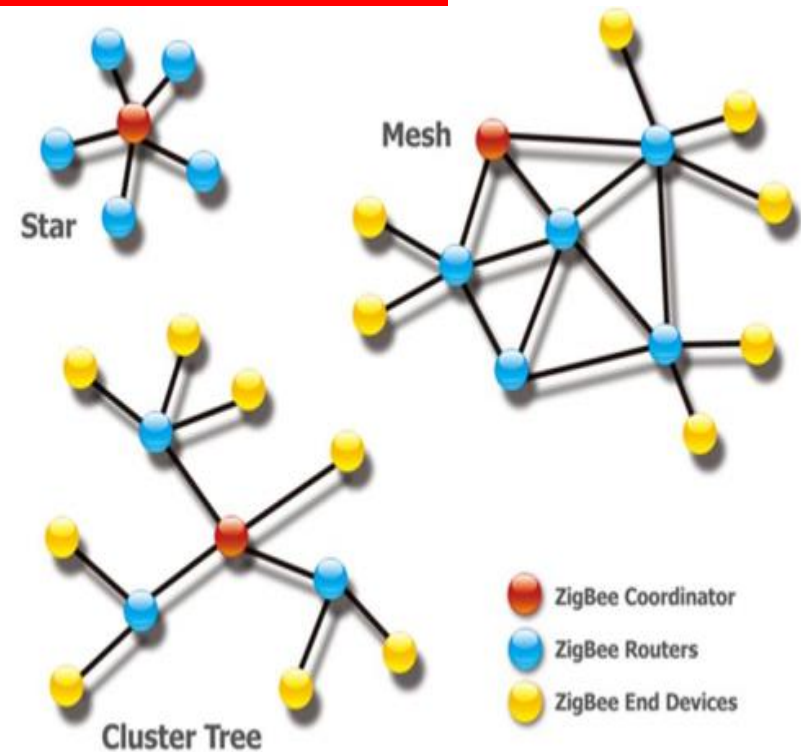
### **Ứng dụng Cloud Computing trong IoT**

Đối với IoT, thường thì đầu não của hệ thống là ở Cloud. Cảm biến và thiết bị thu thập dữ liệu và thực hiện các hành động, nhưng việc xử lý/chỉ huy/phân tích (những thứ thông minh) thường diễn ra ở Cloud.

## 4.3. Các giải pháp công nghệ mạng IoT

### ZigBee (Công nghệ không dây)

- Zigbee là một giao thức được xây dựng theo chuẩn IEEE 802.15.4, phục vụ cho ứng dụng yêu cầu giá thành và công suất thấp, khả năng linh động trong phạm vi rộng.
- Độ trễ truyền tin thấp, tiêu hao, ít năng lượng, giá thành thấp, ít lỗi, dễ mở rộng và thời gian sử dụng pin dài



Hình 4.5: Cấu trúc mạng ZigBee





## 4.3. Các giải pháp công nghệ mạng IoT

### ZigBee (Công nghệ không dây)

#### Quá trình thiết lập trong một mạng Zigbee

- **Quét mạng (Network Scan):** các thiết bị trong mạng sẽ quét các kênh tín hiệu.
- **Thiết lập/Gia nhập mạng:** tạo ra một mạng trên một kênh hoặc gia nhập vào một mạng đã có sẵn.
- **Phát hiện thiết bị:** yêu cầu mạng phát hiện ra địa chỉ của mình trên các kênh được kích hoạt.
- **Phát hiện dịch vụ:** quét các dịch vụ được hỗ trợ trên thiết bị trong phạm vi mạng.
- **Liên kết:** giao tiếp với nhau thông qua các lệnh và các tin nhắn điều khiển.



## 4.3. Các giải pháp công nghệ mạng IoT

### ZigBee (Công nghệ không dây)

#### Ứng dụng ZigBee trong IoT

- ***Tự động hóa nhà***: quản lý nước, kiểm soát ánh sáng
- ***Tự động hóa công nghiệp***: kiểm soát quá trình, quản lý năng lượng
- ***Tự động hóa chăm sóc sức khỏe***: Một người đeo thiết bị ZigBee với cảm biến đo thông số cơ thể thu thập thông tin sức khỏe.



## 4.4. Các giao thức Request/Response

### ***Giao thức HTTP trong IoT***

- HTTP: giao thức yêu cầu/phản hồi không trạng thái, nơi các Client yêu cầu thông tin từ Server và Server sẽ phản hồi các yêu cầu này.
- Giao thức HTTP sử dụng cho dịch vụ Web để truyền dữ liệu dưới dạng văn bản, âm thanh, hình ảnh và video từ Server đến trình duyệt web của người dùng và ngược lại.
- HTTP: nền tảng truyền dữ liệu của ứng dụng duyệt web ngày nay và được sử dụng rộng rãi trong hệ thống IoT.



## 4.4. Các giao thức Request/Response

### *Giao thức HTTP trong IoT*

### Ưu điểm của HTTP

- **Khả năng tìm kiếm:** thực hiện các tìm kiếm SQL và trả về kết quả được định dạng trong tài liệu HTML
- **Dễ lập trình:** HTTP được mã hóa dưới dạng văn bản thuần túy => theo dõi và triển khai dễ hơn các giao thức sử dụng mã yêu cầu tra cứu.
- **Bảo mật:** HTTP 1.0 tải xuống từng tệp qua một kết nối độc lập và sau đó đóng kết nối.



## 4.4. Các giao thức Request/Response

### *Giao thức HTTP trong IoT*

### Nhược điểm của HTTP

- **Không phù hợp với các thiết bị nhỏ:** HTTP quá nặng nên không phù hợp cho các thiết bị này.
- **Không được thiết kế cho giao tiếp dựa trên sự kiện:** HTTP được thiết kế cho giao tiếp dựa trên phản hồi yêu cầu thay vì giao tiếp theo hướng sự kiện.
- **Vấn đề thời gian thực:** Sau khi yêu cầu một tài nguyên đến máy chủ, Client phải đợi Server phản hồi, dẫn đến việc truyền dữ liệu chậm.



## 4.4. Các giao thức Request/Response

### ***Giao thức CoAP trong IoT***

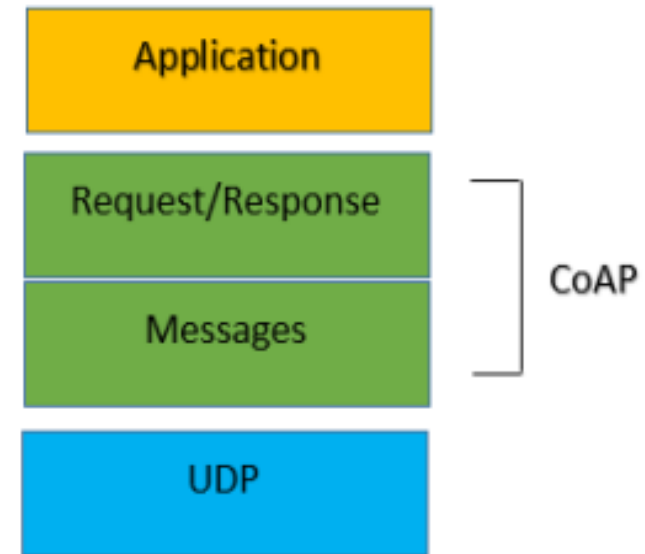
- CoAP (Constrained Application Protocol) là một giao thức đơn giản chi phí thấp được thiết kế riêng cho các thiết bị hiệu năng thấp và có băng thông thấp.
- Giao thức web nhỏ gọn được sử dụng trong M2M;
- Bảo mật bằng DTLS;
- Trao đổi thông điệp không đồng bộ;
- Header gói tin nhỏ, dễ tách thông tin;
- Hỗ trợ URI và loại nội dung;
- Khả năng proxy và bộ nhớ đệm;

## 4.4. Các giao thức Request/Response

### ***Giao thức CoAP trong IoT***

#### **Mô hình cấu trúc CoAP**

- Lớp dưới là lớp bản tin được thiết kế liên quan đến UDP và chuyển tiếp không đồng bộ,
- Lớp yêu cầu/phản hồi liên quan đến phương thức giao tiếp và xử lý bản tin yêu cầu/phản hồi.



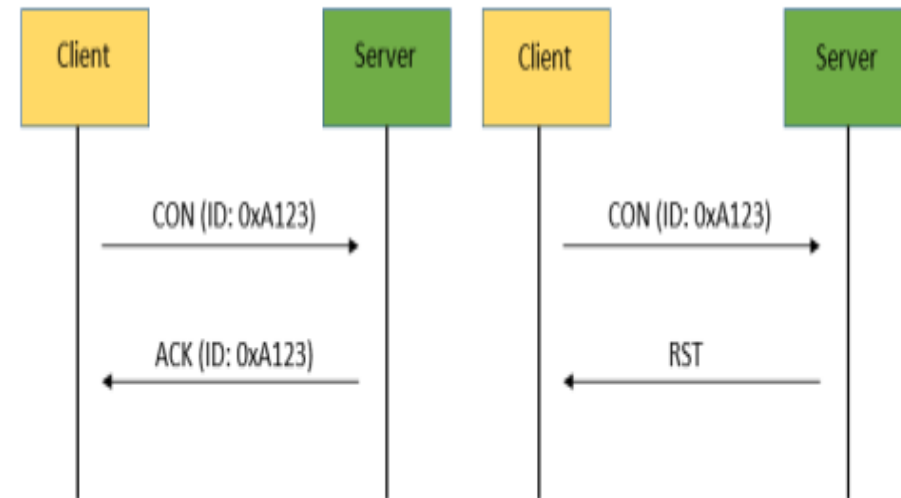
*Hình 4.6: Các lớp của CoAP*

## 4.4. Các giao thức Request/Response

### *Giao thức CoAP trong IoT*

#### Mô hình bản tin CoAP

Một bản tin có thể xác nhận (CON) được truyền đi truyền lại cho đến khi Server gửi lại bản tin xác nhận (ACK) với cùng một ID. Sử dụng thời gian chờ mặc định và giảm thời gian đếm theo cấp số nhân khi truyền bản tin CON. Nếu Server không thể xử lý bản tin truyền đến, nó sẽ phản hồi bằng cách thay thế bản tin xác nhận (ACK) bằng bản tin đặt lại (RST).



Hình 4.7: Truyền bản tin đáng tin cậy

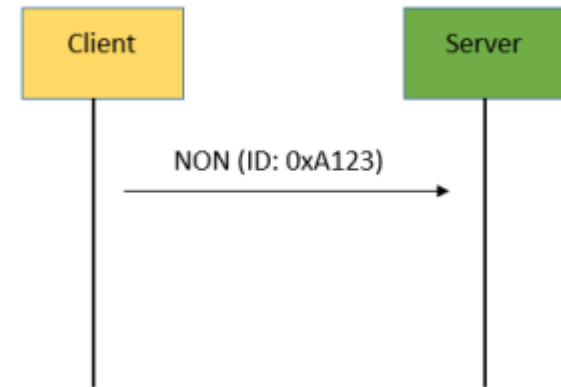


## 4.4. Các giao thức Request/Response

### *Giao thức CoAP trong IoT*

#### Mô hình bản tin CoAP

Truyền tải bản tin không tin cậy:  
Một bản tin không yêu cầu gửi tin cậy, có thể được gửi bằng bản tin không tin cậy. Nó sẽ không được xác nhận, nhưng nó vẫn có ID để phát hiện trùng lặp.



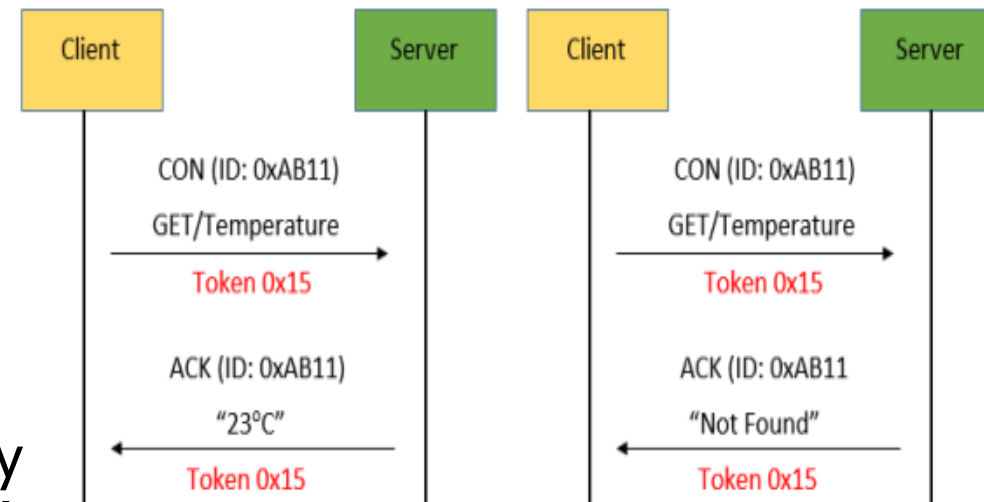
Hình 4.8: Truyền bản tin không tin cậy

## 4.4. Các giao thức Request/Response

### *Giao thức CoAP trong IoT*

#### Mô hình yêu cầu/phản hồi CoAP

Piggy-backed: Một yêu cầu được gửi bằng bản tin có thể xác nhận (CON) hoặc không thể xác nhận (NON), Server sẽ phản hồi bản tin xác nhận (ACK) ngay lập tức cho yêu cầu của Client nếu nó có sẵn. Với phản hồi thành công, ACK chứa mã bản tin phản hồi (được nhận diện bởi mã bản tin), với phản hồi thất bại, ACK chứa mã phản hồi thất bại.



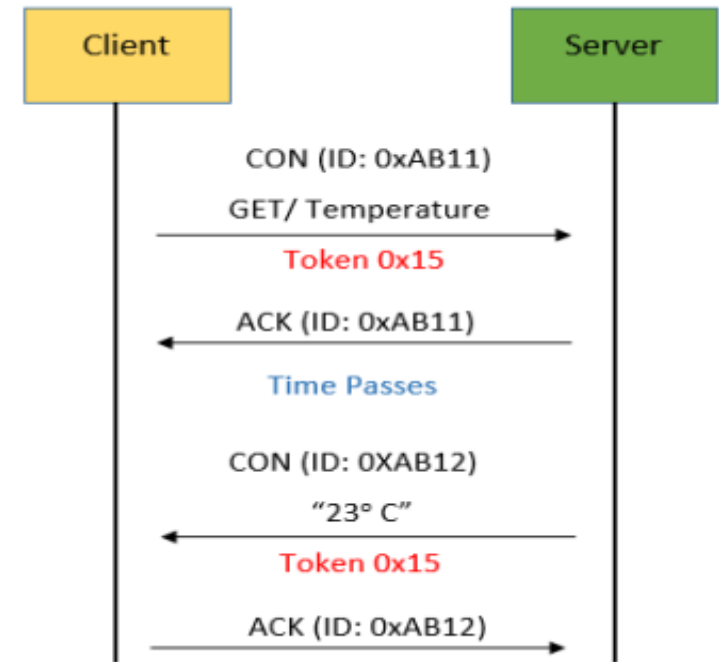
Hình 4.9: Hai yêu cầu GET có phản hồi ngay lập tức, một thành công, một không tìm thấy

## 4.4. Các giao thức Request/Response

### *Giao thức CoAP trong IoT*

### Mô hình yêu cầu/phản hồi CoAP

Phản hồi trì hoãn: Nếu Server nhận được bản tin CON nhưng không thể phản hồi yêu cầu này ngay lập tức, nó sẽ gửi một bản tin ACK. Khi Server sẵn sàng đáp ứng yêu cầu này, nó sẽ gửi một bản tin CON mới đến Client và Client trả lời một bản tin CON kèm theo xác nhận. Bản tin ACK từ Client chỉ để xác nhận bản tin CON từ Server.



Hình 4.10: Yêu cầu GET với phản Hồi trì hoãn



## 4.4. Các giao thức Request/Response

### *Giao thức CoAP trong IoT*

#### Ưu điểm CoAP

- Giao thức đơn giản và header nhỏ gọn hơn do hoạt động qua UDP => tuổi thọ pin dài.
- CoAP có độ trễ thấp hơn so với HTTP;
- CoAP tiêu thụ năng lượng ít hơn so với HTTP;
- CoAP được ứng dụng nhiều trong thiết bị thông tin, thiết bị truyền thông và thiết bị điều khiển trong mạng nhà thông minh.



## 4.4. Các giao thức Request/Response

### *Giao thức CoAP trong IoT*

#### Nhược điểm CoAP

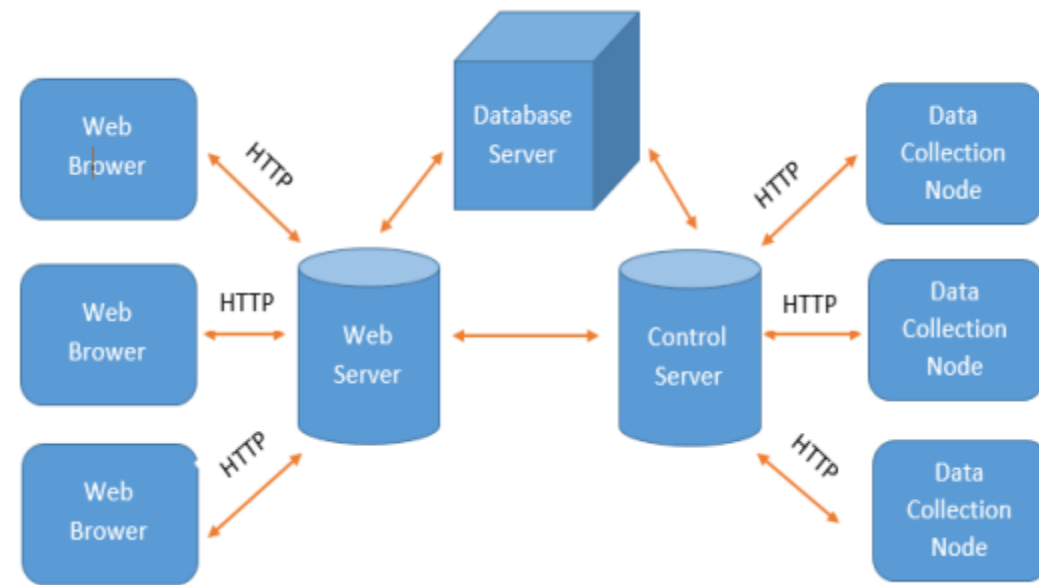
- CoAP là giao thức không tin cậy do sử dụng UDP => các thông điệp CoAP đến không có thứ tự/bị lạc khi chúng đến đích.
- CoAP xác nhận mỗi lần nhận bản tin => tăng thời gian xử lý.
- Giao thức CoAP không được mã hóa như MQTT và sử dụng DTLS để cung cấp bảo mật;

## 4.4. Các giao thức Request/Response

### *Giao thức CoAP trong IoT*

### Ứng dụng CoAP cho nhà thông minh

- Nhận ra cảnh báo mất an toàn, điều khiển từ xa và tiết kiệm năng lượng chủ động.
- Mọi nút thu thập dữ liệu với Client, CoAP có thể trao đổi thông tin với các nút khác.
- CoAP được cài đặt trong mạng LAN hoặc Internet.

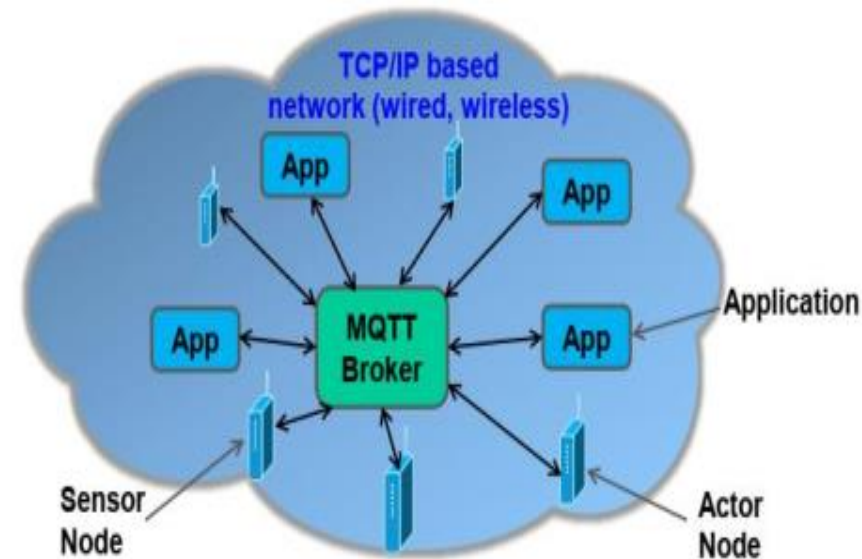


Hình 4.11: Sơ đồ hệ thống điều khiển năng lượng

## 4.5. Các giao thức Publish/Subscriber

### Giao thức MQTT

- Giao thức MQTT là một giao thức truyền thông điệp mở, và nhỏ gọn.
- Hoạt động theo mô hình Xuất bản/Theo dõi (Publish/Subscribe) sử dụng băng thông thấp, độ tin cậy cao, có khả năng hoạt động trong điều kiện đường truyền không ổn định phù hợp với các thiết bị có tài nguyên giới hạn.

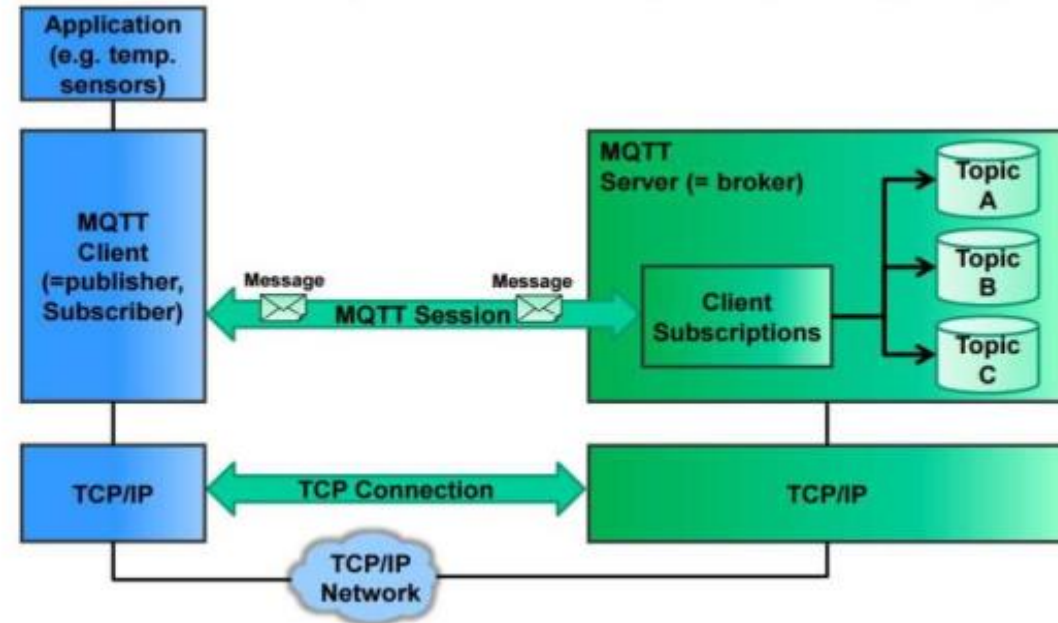


Hình 4.12: Giao thức MQTT

## 4.5. Các giao thức Publish/Subscriber

### Giao thức MQTT

- MQTT Client, MQTT Broker, phiên MQTT, khối đăng kí, và các chủ đề.
- MQTT được coi là giao thức định hướng bản tin, sử dụng các "chủ đề" thay thế cho địa chỉ (IP, cổng) mà bản tin được gửi đến.



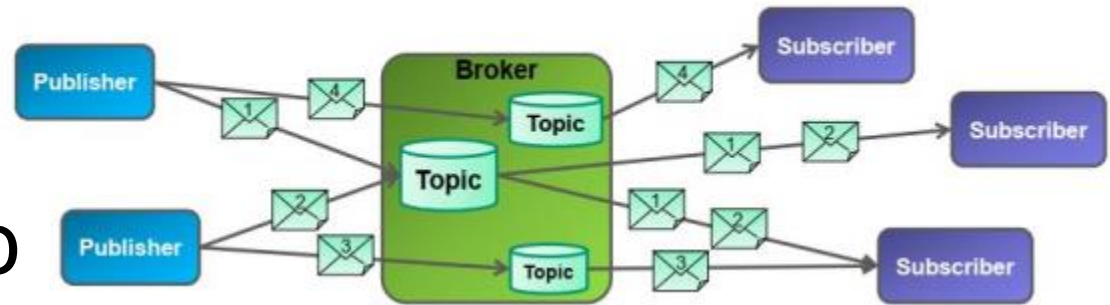
Hình 4.13: Kiến trúc giao thức MQTT



## 4.5. Các giao thức Publish/Subscriber

### Giao thức MQTT

Một thiết bị có thể theo dõi nhiều chủ đề khác nhau, và nó sẽ nhận được tất cả các bản tin được xuất bản tới các chủ đề đó.



Hình 4.14: Mô hình Xuất bản/Theo dõi

## 4.5. Các giao thức Publish/Subscriber

### Giao thức MQTT

#### (1) MQTT Client

- Một MQTT Client có thể là một thiết bị bất kỳ có khả năng gửi và nhận dữ liệu, (cảm biến, server).
- Có 2 loại MQTT Client là "Publisher" và "Subscriber". Trong đó, thiết bị đóng vai trò là publisher có khả năng xuất bản các bản tin, ngược lại, thiết bị đóng vai trò là subscriber có khả năng nhận các bản tin thuộc chủ đề mà nó theo dõi.



Hình 4.15: Publisher và Subscriber



## 4.5. Các giao thức Publish/Subscriber

### Giao thức MQTT

- **(2) MQTT Broker**

MQTT Broker: thiết bị trung tâm chịu trách nhiệm xử lý truyền thông giữa các MQTT Client và phân phát các bản tin giữa chúng dựa vào các chủ đề mà chúng theo dõi.

- Một Broker có khả năng xử lý hàng ngàn thiết bị kết nối tới nó tại cùng một thời điểm.
- Dựa trên bản tin nhận được, Broker sẽ xác định và tìm kiếm tất cả các thiết bị đã theo dõi tới chủ đề của bản tin đó và sẽ chuyển các bản tin tới các thiết bị đó.



## 4.5. Các giao thức Publish/Subscriber

### Giao thức MQTT

#### ***(3) Phiên MQTT***

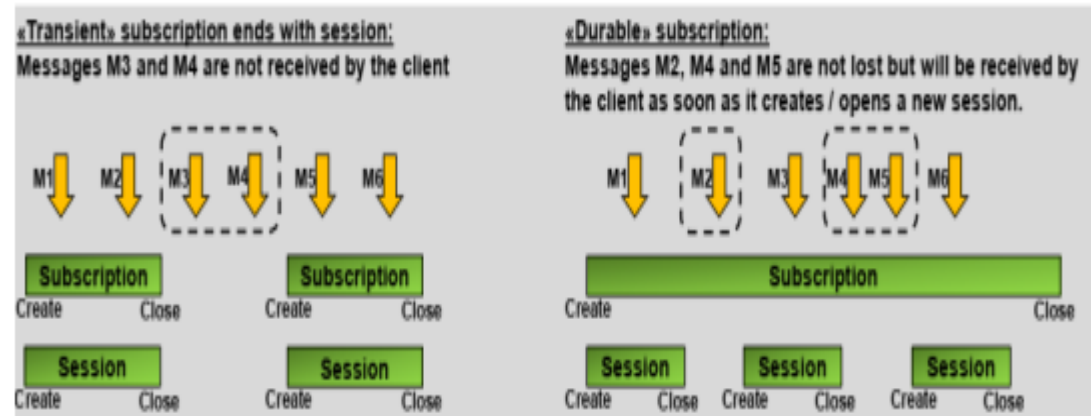
- Một phiên MQTT: toàn bộ quá trình Client được kết nối đến Server.
- Tất cả các hoạt động truyền thông giữa Client và Server đều được diễn ra trong phiên MQTT.
- Có hai loại phiên MQTT: thường trực và không thường trực được bởi thiết lập cờ *cleanSession*.

## 4.5. Các giao thức Publish/Subscriber

### Giao thức MQTT

#### (4) *Khôi đăng ký*

- Khi đăng ký trên Broker sẽ làm nhiệm vụ gán Client đến chủ đề tương ứng mà Client đăng ký theo dõi => khi Client có thể nhận tất cả các bản tin thuộc chủ đề đó.



Hình 4.16: Các loại đăng ký trong Giao thức MQTT

## 4.5. Các giao thức Publish/Subscriber

### Giao thức MQTT

#### (5) Chủ đề

- Là một hàng đợi các bản tin có cùng đặc điểm chung nào đó.
- Mỗi chủ đề có thể chứa nhiều cấp chủ đề, mỗi cấp chủ đề được phân tách bởi dấu gạch chéo “/”



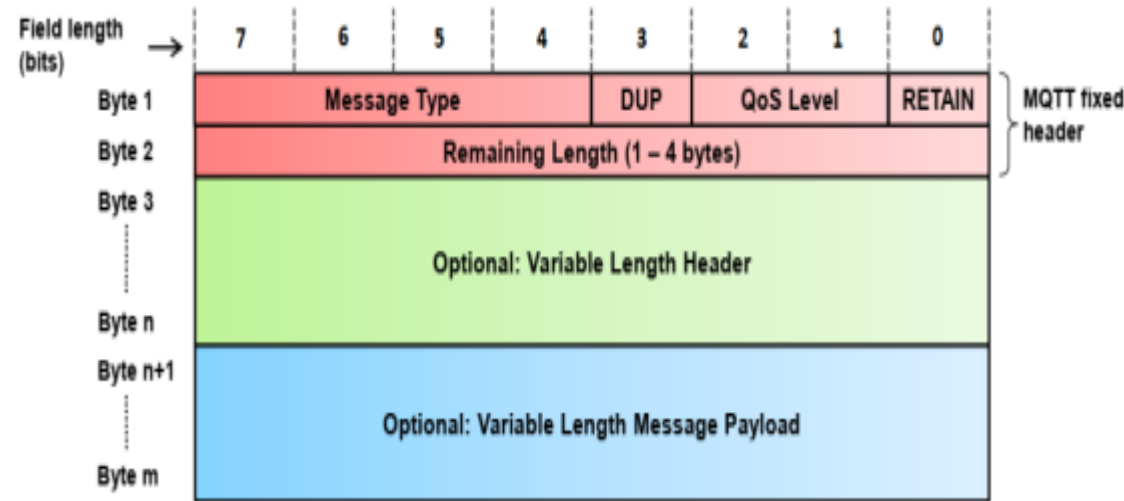
Hình 4.17: Định dạng phân cấp của chủ đề

## 4.5. Các giao thức Publish/Subscriber

### Giao thức MQTT

#### *Các gói tin điều khiển trong giao thức MQTT*

- Tiêu đề có độ dài cố định (2 bytes)
- Tiêu đề có độ dài thay đổi
- Payload



Hình 4.18: Định dạng của gói tin điều khiển

## 4.5. Các giao thức Publish/Subscriber

### Giao thức MQTT

#### *Các gói tin điều khiển trong giao thức MQTT*

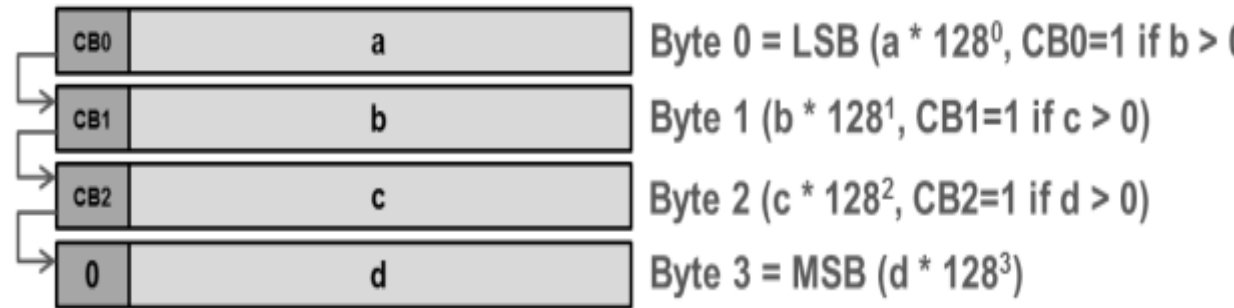
Loại gói tin	Giá trị (thập phân)	Luồng hoạt động	Mô tả
Reserved	0	Chưa dùng	Chưa dùng
CONNECT	1	Client → Broker	Client yêu cầu kết nối tới Broker
CONNACK	2	Broker → Client	Kết nối được chấp nhận
PUBLISH	3	Publisher → Broker Broker → Subscriber	Xuất bản bản tin
PUBACK	4	Publisher → Broker Broker → Subscriber	Việc xuất bản bản tin được chấp nhận
PUBREC	5	Publisher → Broker Broker → Subscriber	Gói tin PUBLISH đã được nhận (quá trình chuyển phát lần 1)



## 4.5. Các giao thức Publish/Subscriber

### Giao thức MQTT

Độ dài còn lại của gói tin được tính theo công thức sau:  $a \cdot 128^0 + b \cdot 128^1 + c \cdot 128^2 + d \cdot 128^3$ .



Hình 4.19: Cách tính giá trị trường độ dài còn lại

**Ví dụ 1:** Độ dài còn lại của gói tin (RL) là 364 bit sẽ được biểu diễn như sau:

$$RL = 108 \cdot 128^0 + 2 \cdot 128^1 = 108 + 256, \text{ trong đó, } a = 108, \text{ CB0} = 1, b = 2, \text{ CB1} = 0.$$

Do đó, 2 byte của RL là: 0xEC (byte 2), 0x02 (byte 3).

**Ví dụ 2:** Độ dài còn lại của gói tin (RL) là 25897 bit sẽ được biểu diễn như sau:

$$RL = 41 \cdot 128^0 + 74 \cdot 128^1 + 1 \cdot 128^2 = 41 + 9.472 + 16.384 = 25.897,$$

( $a = 41, \text{ CB0} = 1, b = 74, \text{ CB1} = 1, c = 1, \text{ CB2} = 0$ ).

Do đó, 3 byte của RL là: 0xA9 (byte 1), 0xCA (byte 2), 0x01 (byte 3).

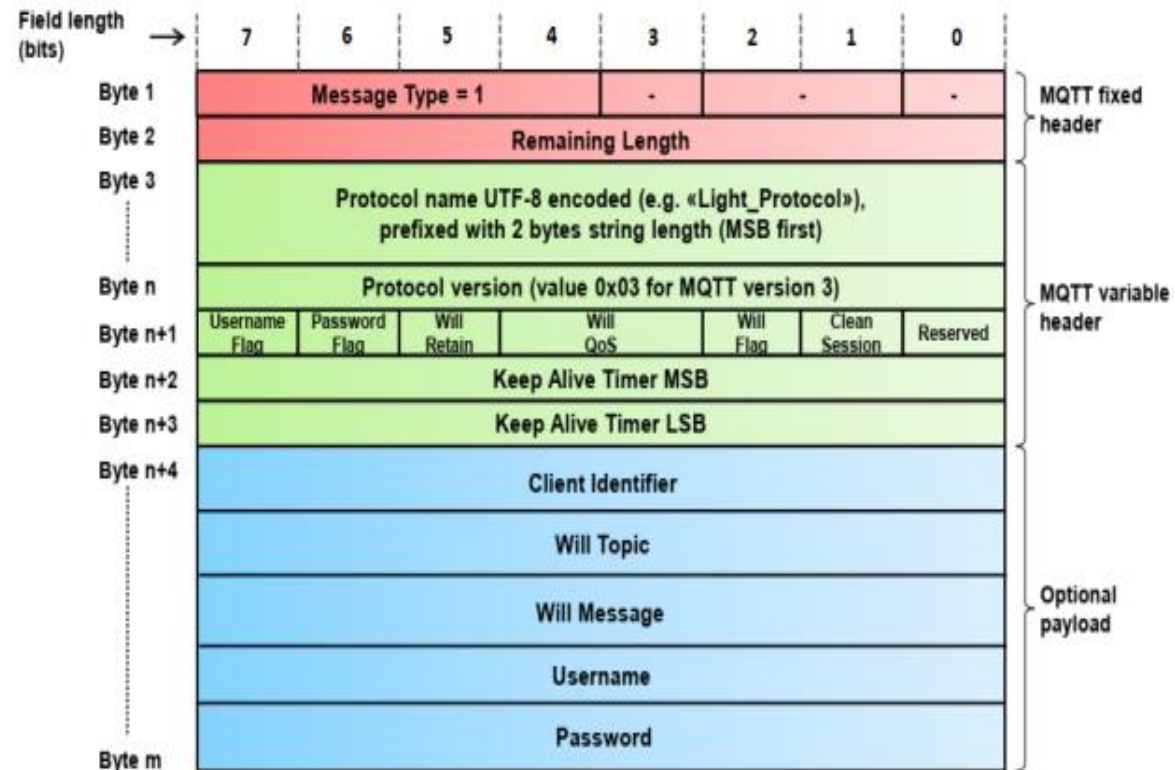
## 4.5. Các giao thức Publish/Subscriber

### Giao thức MQTT

#### *Gói tin điều khiển trong MQTT*

- Gói tin CONNECT*

Trường khai báo tên giao thức, trường khai báo phiên bản MQTT sử dụng, các cờ kết nối, và trường khai báo thời gian Keep Alive



Hình 4.20: Gói tin Connect



## 4.5. Các giao thức Publish/Subscriber

### Giao thức MQTT

#### ***Gói tin điều khiển trong MQTT***

- *Gói tin CONNECT* : hoạt động

Khi một kết nối TCP/IP được thiết lập từ Client đến Broker, thì một phiên ở tầng MQTT cũng được tạo ra bằng việc Client gửi một gói tin CONNECT đến Broker.

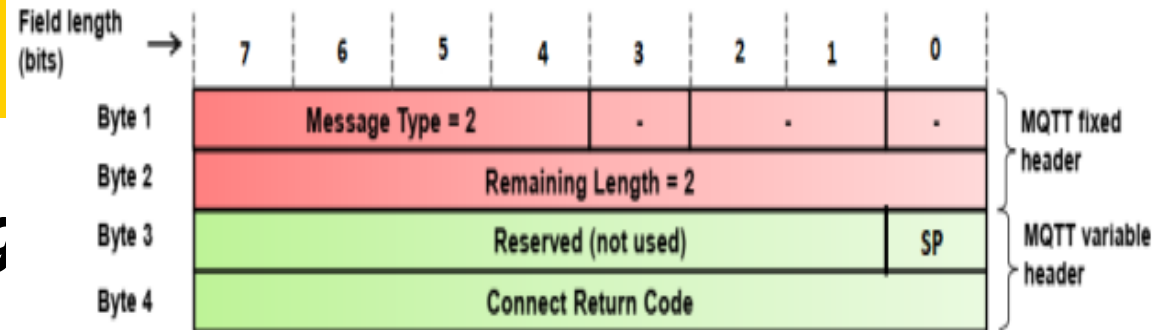
=> Broker sẽ gửi gói tin CONNACK để trả lời gói tin CONNECT từ Client

## 4.5. Các giao thức Publish/Subscriber

### Giao thức MQTT

#### *Gói tin điều khiển trong MQTT*

- *Gói tin CONNACK* :  
được gửi bởi Broker để phản hồi lại gói tin CONNECT từ Client.  
=> gói tin đầu tiên được gửi từ Broker đến Client



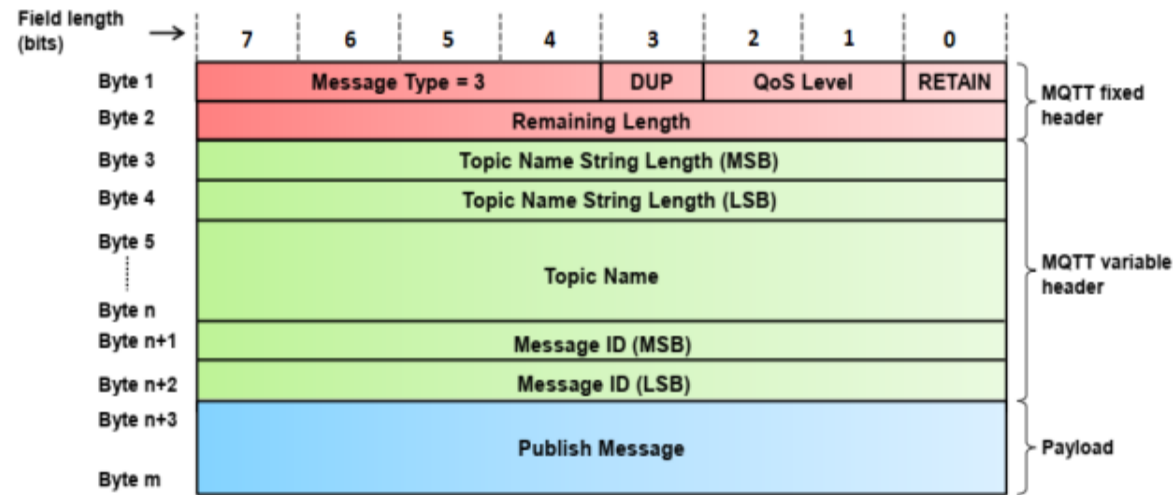
Hình 4.21: Gói tin CONNACK

## 4.5. Các giao thức Publish/Subscriber

### Giao thức MQTT

#### *Gói tin điều khiển trong MQTT*

- *Gói tin PUBLISH* :  
chuyển bản tin MQTT từ Client (Publisher) đến Broker hoặc từ Broker đến Client (Subscriber).



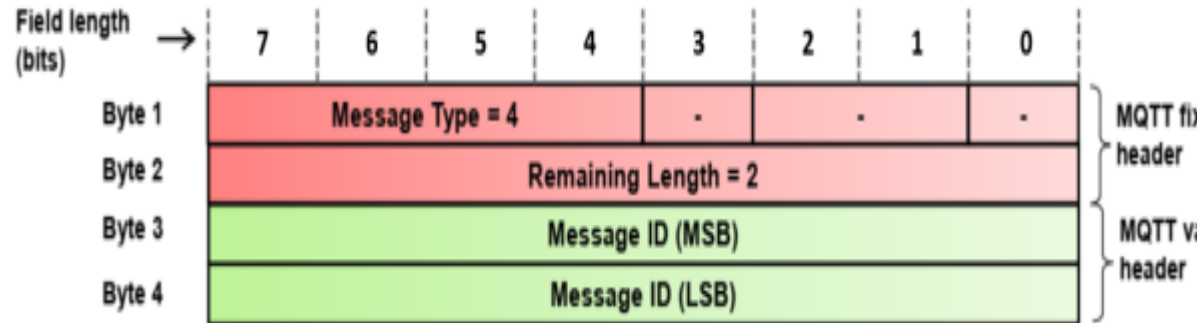
Hình 4.22: Gói tin Publish

## 4.5. Các giao thức Publish/Subscriber

### Giao thức MQTT

#### *Gói tin điều khiển trong MQTT*

- *Gói tin PUBACK* : sử dụng bởi bên nhận để phản hồi lại gói tin PUBLISH.
- Gói tin PUBACK không chứa phần dữ liệu.



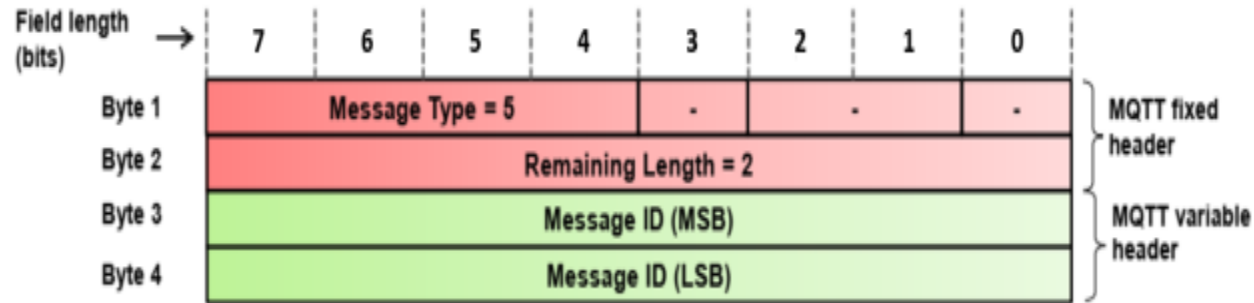
Hình 4.23: Gói tin Puback

## 4.5. Các giao thức Publish/Subscriber

### Giao thức MQTT

#### *Gói tin điều khiển trong MQTT*

- *Gói tin PUBREC*: được sử dụng bởi bên nhận để phản hồi lại gói tin PUBLISH có QoS mức 2



Hình 4.24: Gói tin PUBREC

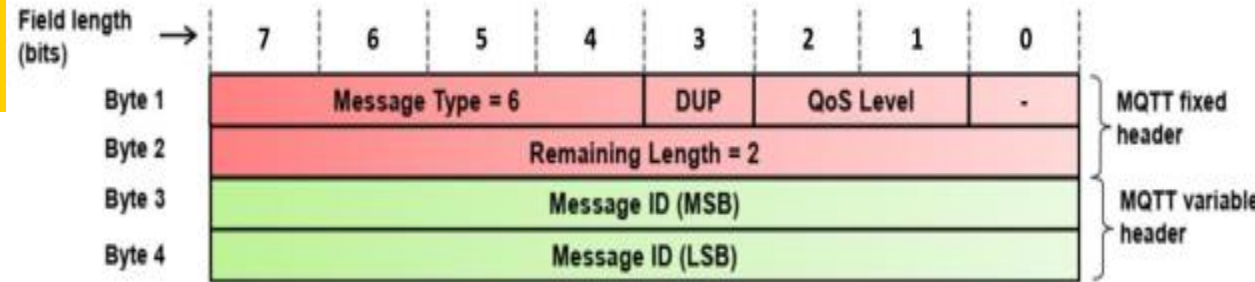
## 4.5. Các giao thức Publish/Subscriber

### Giao thức MQTT

#### *Gói tin điều khiển trong MQTT*

- *Gói tin PUBREL*: được sử dụng bởi bên gửi để phản hồi lại gói tin PUBREC.

=> gói tin thứ 3 trong quy trình xuất bản bản tin có QoS mức 2.



Hình 4.25: Gói tin PUBREL

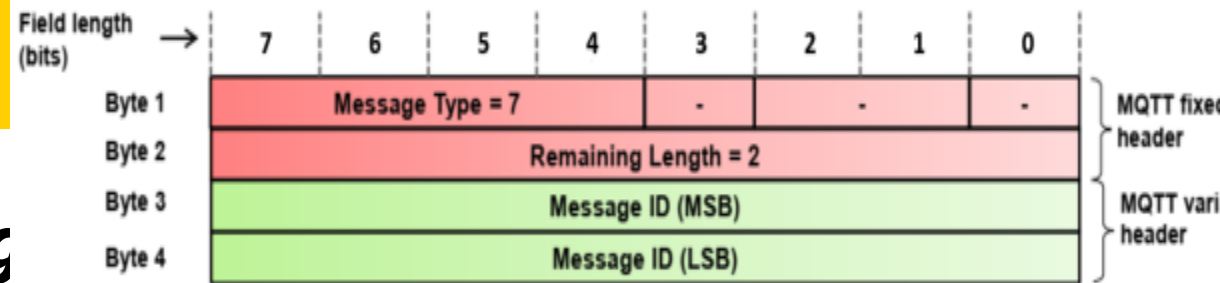


## 4.5. Các giao thức Publish/Subscriber

### Giao thức MQTT

#### Gói tin điều khiển trong MQTT

- *Gói tin PUBCOMP*: được sử dụng bởi bên nhận để phản hồi gói tin PUBREL.  
=> gói tin thứ 4 trong quá trình xuất bản bản tin với QoS mức 2



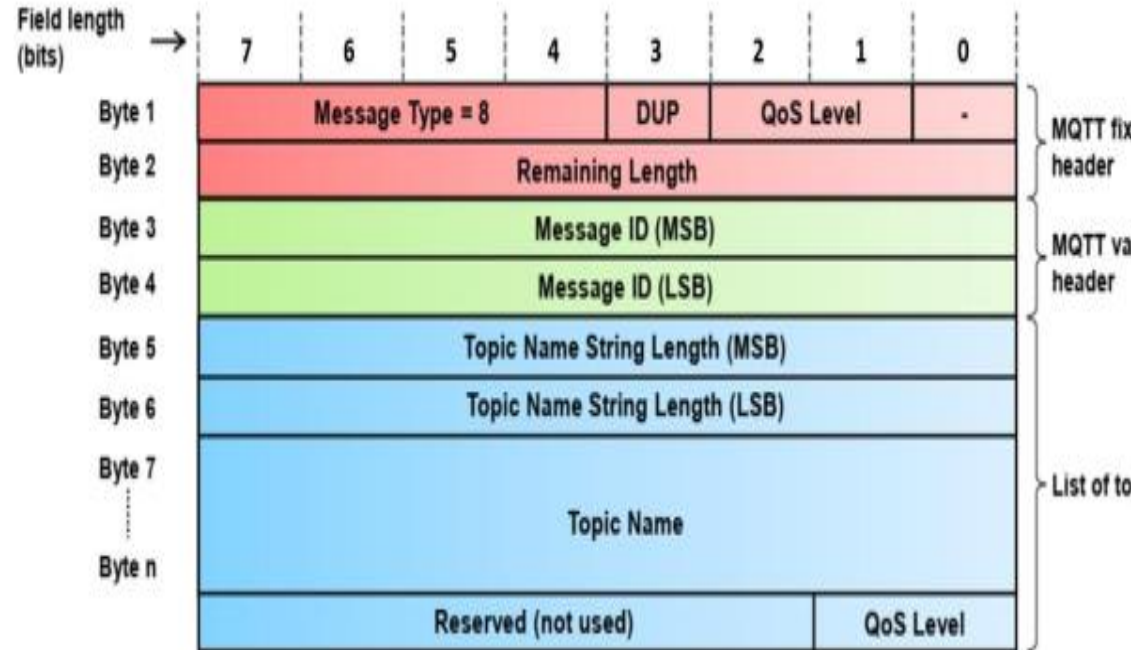
Hình 4.26: Gói tin PUBCOMP

## 4.5. Các giao thức Publish/Subscriber

### Giao thức MQTT

#### *Gói tin điều khiển trong MQTT*

- *Gói tin SUBSCRIBE*: được gửi từ Client đến Broker để tạo một hoặc nhiều đăng kí theo dõi.



Hình 4.27: Gói tin SUBSCRIBE



## 4.5. Các giao thức Publish/Subscriber

### Giao thức MQTT

#### *Gói tin điều khiển trong MQTT*

- *Gói tin UNSUBSCRIBE*: được gửi bởi Client đến Broker để huỷ đăng kí theo dõi tới các chủ đề.
- *Gói tin UNSUBACK*: được gửi bởi Broker đến Client để xác nhận việc nhận gói tin UNSUBSCRIBE.



## 4.5. Các giao thức Publish/Subscriber

### Giao thức MQTT

***Các quy trình hoạt động của giao thức MQTT***

- *Thiết lập kết nối và đăng kí theo dõi*
- *Xuất bản bản tin MQTT*



## 4.5. Các giao thức Publish/Subscriber

### Giao thức MQTT-SN

- Giao thức MQTT-SN sử dụng mô hình “Xuất bản/Theo dõi” được thiết kế dành riêng cho mạng cảm biến không dây.
- MQTT-SN là biến thể của giao thức MQTT có thể giải quyết được các đặc tính riêng của các mạng cảm biến không dây như nhiễu fading và tốc độ truyền thông thấp.
- Thiết kế để dùng trong tất cả các mạng mà có truyền thông 2 chiều giữa các nút mạng bất kỳ và một nút mạng đặc biệt (Gateway).



## 4.5. Các giao thức Publish/Subscriber

### Giao thức MQTT-SN

#### So sánh MQTT-SN và MQTT

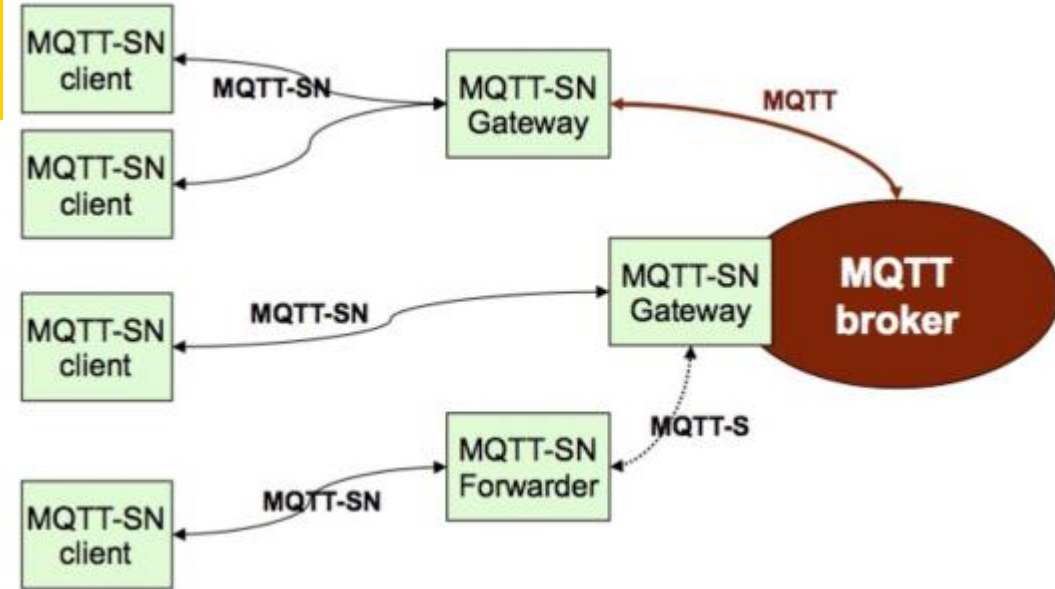
- Gói tin CONNECT được chia thành ba gói tin nhỏ hơn. Hai gói tin thêm là tùy chọn và được sử dụng để truyền chủ đề WILL và bản tin WILL đến server.
- Để giải quyết các vấn đề độ dài bản tin ngắn và băng thông giới hạn của các mạng WSN.
- TopicID định sẵn, tên chủ đề rút gọn được giới thiệu trong giao thức MQTT-SN giúp cho Client không cần thực hiện khai báo tên chủ đề.

## 4.5. Các giao thức Publish/Subscriber

### Giao thức MQTT-SN

#### Kiến trúc giao thức MQTT-SN

- MQTT-SN Client,
- MQTT-SN gateway (GW)
- MQTT-SN forwarder.



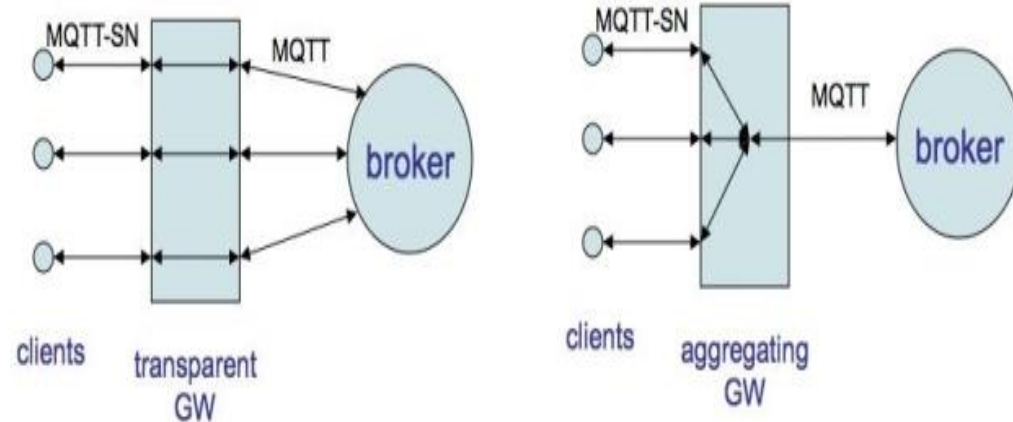
Hình 4.28: Kiến trúc giao thức MQTT-SN

## 4.5. Các giao thức Publish/Subscriber

### Giao thức MQTT-SN

#### ***Phân loại GW trong MQTT***

- Transparent GW
- Aggregating GW



*Hình 4.29: Transparent Gateway và Aggregating Gateway*



## 4.5. Các giao thức Publish/Subscriber

### Giao thức MQTT-SN

#### Định dạng gói tin điều khiển trong MQTT-SN

Trong giao thức MQTT-SN, một số gói tin mới được sử dụng để thực hiện các tính năng không có trong giao thức MQTT như: Quảng bá và tìm kiếm Gateway

Phần tiêu đề gói tin (2 hoặc 4 octet)	Phần thay đổi của gói tin (n octets)
--	---

Hình 4.30: Định dạng gói tin trong MQTT-SN



## 4.5. Các giao thức Publish/Subscriber

### Giao thức MQTT-SN

#### Một số tính năng của MQTT-SN

- *Quảng bá và tìm kiếm Gateway*
- *Thiết lập kết nối phía Client*
- *Clean session*
- *Tiến trình cập nhật dữ liệu bản tin WILL*
- *Tiến trình khai báo tên chủ đề*
- *Tiến trình xuất bản của Client*
- *TopicID định sẵn và tên chủ đề rút gọn*
- *Tiến trình xuất bản bản tin với mức QoS là -1*



## 4.5. Các giao thức Publish/Subscriber

### Giao thức MQTT-SN

#### Một số tính năng của MQTT-SN

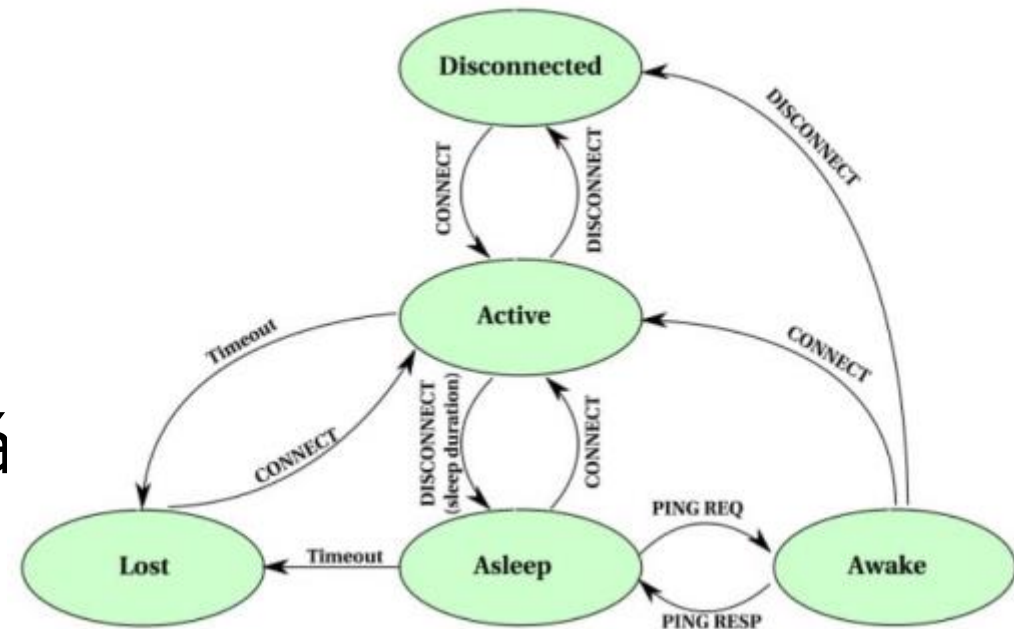
- *Tiến trình đăng kí và huỷ đăng kí theo dõi chủ đề của Client*
- *Tiến trình xuất bản của GW*
- *Tiến trình PING và Keep Alive*
- *Tiến trình huỷ kết nối của Client*
- *Tiến trình truyền lại bản tin của Client*

## 4.5. Các giao thức Publish/Subscriber

### Giao thức MQTT

#### ***Tính năng MQTT-SN***

- Giao thức MQTT-SN cho phép Client rơi vào trạng thái ngủ khi chúng không hoạt động và có khả năng thức dậy bất cứ khi nào chúng có dữ liệu phải gửi đi.  
=> Tiết kiệm năng lượng



Hình 4.31: Sơ đồ chuyển trạng thái của Client



## Kết luận chương

---

### Kiến trúc, công nghệ IoT:

- Kiến trúc IoT
- Đặc điểm IoT
- Ứng dụng IoT
- Tiêu chuẩn trong IoT

### Giao thức trong IoT

- Request/ Response (CoAP, HTTP)
- Publish/Subscriber (MQTT, MQTT-SN)