# BÁO CÁO BÀI TẬP LỚN / ĐỒ ÁN MÔN HỌC

## MÔN HỌC: HỆ ĐIỀU HÀNH THỜI GIAN THỰC (RTOS) & LẬP TRÌNH NHÚNG

**ĐỀ TÀI:**

### THIẾT KẾ HỆ THỐNG CẢNH BÁO KHOẢNG CÁCH VÀ XÁC THỰC BẢO MẬT KHÔNG DÂY (BLE) SỬ DỤNG FREERTOS TRÊN ESP32

Sinh viên thực hiện: Hà Quang Chương

Mã sinh viên: 22810540016

Lớp: D17DT&KTMT1

## MỤC LỤC

1. [Tổng quan đề tài](https://www.google.com/search?q=%231-t%E1%BB%95ng-quan-%C4%91%E1%BB%81-t%C3%A0i)
2. [Cơ sở lý thuyết](https://www.google.com/search?q=%232-c%C6%A1-s%E1%BB%9F-l%C3%BD-thuy%E1%BA%BFt)
3. [Thiết kế hệ thống](https://www.google.com/search?q=%233-thi%E1%BA%BFt-k%E1%BA%BF-h%E1%BB%87-th%E1%BB%91ng)
4. [Thiết kế phần mềm (Software Architecture)](https://www.google.com/search?q=%234-thi%E1%BA%BFt-k%E1%BA%BF-ph%E1%BA%A7n-m%E1%BB%81m-software-architecture)
5. [Kết quả thực nghiệm](https://www.google.com/search?q=%235-k%E1%BA%BFt-qu%E1%BA%A3-th%E1%BB%B1c-nghi%E1%BB%87m)
6. [Kết luận & Hướng phát triển](https://www.google.com/search?q=%236-k%E1%BA%BFt-lu%E1%BA%ADn--h%C6%B0%E1%BB%9Bng-ph%C3%A1t-tri%E1%BB%83n)

## 1. TỔNG QUAN ĐỀ TÀI

### 1.1. Đặt vấn đề

Trong các ứng dụng IoT hiện đại như chìa khóa thông minh (Smart Key), thẻ nhân viên an ninh, hoặc hệ thống chống trộm, các yêu cầu kỹ thuật cốt lõi bao gồm:

* **Xác thực không dây:** Hệ thống phải phân biệt được thiết bị "chính chủ" và thiết bị giả mạo để đảm bảo an ninh.
* **Ước lượng khoảng cách:** Nhận biết ngữ cảnh khi người dùng lại gần hoặc đi xa để đưa ra hành động tự động (ví dụ: tự động mở cửa, tự động khóa xe).
* **Tiết kiệm năng lượng & Đa nhiệm:** Thiết bị phải xử lý nhiều tác vụ cùng lúc (quét sóng, tính toán mã hóa, hiển thị LED) mượt mà, không bị treo (blocking) và tối ưu hóa năng lượng.

### 1.2. Mục tiêu kỹ thuật

Đồ án tập trung xây dựng hệ thống gồm 2 thiết bị sử dụng vi điều khiển ESP32:

1. **Thiết bị Tag (Trạm Phát):** Hoạt động bí mật, phát tín hiệu quảng bá chứa thông tin bảo mật liên tục.
2. **Thiết bị Station (Trạm Thu):** Thực hiện quét tín hiệu, xác thực mã băm MD5, đo cường độ sóng (RSSI) và đưa ra cảnh báo bằng đèn LED với các nhịp điệu khác nhau tùy theo khoảng cách vật lý.

## 2. CƠ SỞ LÝ THUYẾT

### 2.1. Công nghệ BLE (Bluetooth Low Energy) và chế độ Advertising

Hệ thống không sử dụng cơ chế kết nối "Pairing/Bonding" truyền thống mà sử dụng chế độ **Broadcasting (Quảng bá)**.

* **Trạm Phát (Advertiser):** Đóng vai trò như một ngọn hải đăng (Beacon), liên tục phát gói tin chứa dữ liệu định danh ra môi trường xung quanh (Advertising Data).
* **Trạm Thu (Scanner):** Lắng nghe và lọc các gói tin này mà không cần thiết lập kết nối hai chiều, giúp tiết kiệm năng lượng và tăng tốc độ phản hồi.

### 2.2. Đo khoảng cách bằng RSSI

**RSSI (Received Signal Strength Indicator)** là chỉ số cường độ tín hiệu thu được, đơn vị là dBm (thường là số âm).

* **Nguyên lý:** Sóng điện từ suy hao theo khoảng cách. RSSI càng lớn (gần 0) thì thiết bị càng gần. RSSI càng nhỏ (càng âm nhiều) thì thiết bị càng xa.
* **Các ngưỡng thiết lập trong đồ án:**
  + **Ngưỡng GẦN:** RSSI > -60 dBm (Khoảng cách < 1m).
  + **Ngưỡng XA:** RSSI < -75 dBm (Khoảng cách > 4m).
  + **Vùng Đệm (Hysteresis):** -75 dBm đến -60 dBm (Giữ trạng thái cũ để tránh nhiễu).

### 2.3. Hệ điều hành FreeRTOS

Sử dụng FreeRTOS để quản lý tài nguyên và lập lịch cho vi điều khiển:

* **Task (Tác vụ):** Chia nhỏ chương trình thành các luồng độc lập (Quét, Xử lý, Hiển thị).
* **Queue (Hàng đợi):** Dùng để truyền dữ liệu an toàn giữa các Task (từ Task Quét sang Task Xử lý) mà không gây xung đột vùng nhớ.

### 2.4. Thuật toán bảo mật MD5 & Rolling Code

Để chống lại việc sao chép tín hiệu (Replay Attack), hệ thống sử dụng cơ chế Mã hóa động:

1. **Shared Secret (Khóa bí mật):** Chuỗi ký tự chỉ 2 thiết bị biết, được nạp cứng vào firmware, không bao giờ gửi qua sóng.
2. **Salt (Muối):** Một số ngẫu nhiên sinh ra mỗi 2 giây tại Trạm Phát.
3. Signature (Chữ ký số): Kết quả của hàm băm một chiều:  
   $$Signature = MD5(Salt + SecretKey)$$  
     
   Trạm Thu khi nhận được Salt sẽ tự tính lại MD5 để so sánh xác thực.

## 3. THIẾT KẾ HỆ THỐNG

### 3.1. Sơ đồ luồng dữ liệu (Data Flow)

A diagram of a system

AI-generated content may be incorrect.

### 3.2. Kịch bản hoạt động (Use Case)

* **Trạm Phát:** Hoạt động âm thầm, không nháy đèn (chế độ "Stealth") để bảo mật vị trí và tiết kiệm pin tối đa.
* **Trạm Thu:**
  + Nhận tín hiệu Kiểm tra chữ ký MD5.
  + Nếu **Sai chữ ký**: Bỏ qua gói tin (coi như rác/tấn công).
  + Nếu **Đúng chữ ký**: Kiểm tra RSSI.
    - **Rất gần (< 1m):** Đèn LED chớp tắt liên hồi (Strobe Light) cảnh báo khẩn cấp/mở khóa.
    - **Xa (3-5m):** Đèn LED nháy chậm rãi (Beacon Mode) báo hiệu đã nhận diện.
    - **Mất tín hiệu / Vùng đệm:** Tắt đèn.

## 4. THIẾT KẾ PHẦN MỀM (SOFTWARE ARCHITECTURE)

Hệ thống được xây dựng trên nền tảng FreeRTOS để đảm bảo tính thời gian thực (Real-time).

### 4.1. Cấu trúc Trạm Phát (Tag) - "Chế độ Ninja"

Thiết kế tối giản với 1 Task duy nhất hoặc chạy trong loop() để tối ưu năng lượng deep-sleep (nếu cần).

* **Logic:**
  1. Sinh số ngẫu nhiên (Salt).
  2. Tạo chữ ký MD5 từ Salt + Key.
  3. Cập nhật gói tin quảng bá (Manufacturer Data).
  4. Phát sóng.
* **Lưu ý:** Tắt toàn bộ đèn LED trên mạch Tag.

### 4.2. Cấu trúc Trạm Thu (Station) - Kiến trúc Đa nhiệm

Trạm thu sử dụng mô hình **Producer-Consumer** với Hàng đợi (Queue):

|  |  |  |
| --- | --- | --- |
| **Tên Task** | **Độ ưu tiên** | **Chức năng** |
| **Task\_Scan** | 1 | Quét môi trường liên tục (Scan Window = 100ms). Lọc thiết bị theo UUID. Đẩy dữ liệu thô vào dataQueue. |
| **Task\_Process** | 2 | Lấy dữ liệu từ dataQueue. Tính toán MD5 xác thực. So sánh RSSI để cập nhật biến toàn cục currentState. |
| **Task\_Indication** | 1 | Đọc currentState để điều khiển đèn LED theo nhịp điệu tương ứng (Sử dụng vTaskDelay độc lập). |

### 4.3. Máy trạng thái (State Machine) điều khiển đèn LED

Mã nguồn của Task\_Indication xử lý hiển thị tách biệt hoàn toàn với logic xử lý dữ liệu, giúp đèn nháy mượt mà không bị khựng (blocking) khi hệ thống đang bận tính toán MD5.

**Trích đoạn code thực tế:**

A computer screen shot of a program

AI-generated content may be incorrect.

## 5. KẾT QUẢ THỰC NGHIỆM

### 5.1. Môi trường kiểm thử

Thiết bị được cấp nguồn qua cổng USB, quan sát log qua Serial Monitor và trạng thái đèn LED thực tế.

### 5.2. Kết quả đo đạc & Phản hồi hệ thống

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Kịch bản** | **Thao tác thực hiện** | **Kết quả quan sát (Trạm Thu)** | **Đánh giá** |
| 1 | **Bảo mật** | Đặt Trạm Phát cạnh Trạm Thu. | Serial Monitor hiện Salt thay đổi, tính toán MD5 khớp. Đèn bắt đầu hoạt động. | **Đạt** |
| 2 | **Cảnh báo GẦN** | Khoảng cách < 1 mét (RSSI > -60). | Đèn LED chớp tắt rất nhanh, liên tục (Strobe effect). | **Đạt** |
| 3 | **Cảnh báo XA** | Khoảng cách > 4 mét (RSSI < -75). | Đèn LED chuyển sang nhịp: Sáng 1 giây, Tắt 1 giây. | **Đạt** |
| 4 | **Vùng đệm** | Khoảng cách 2-3 mét (-60 đến -75). | Đèn LED giữ nguyên trạng thái hoặc tắt (tránh hiện tượng nháy loạn). | **Đạt** |
| 5 | **Mất tín hiệu** | Rút nguồn Trạm Phát. | Sau 3 giây (Timeout), đèn LED tắt hoàn toàn. Trạng thái về IDLE. | **Đạt** |

### 5.3. Nhận xét

* Hệ thống hoạt động ổn định, không xảy ra hiện tượng xung đột tài nguyên hay treo vi điều khiển nhờ sử dụng cơ chế Queue và Mutex ngầm của FreeRTOS.
* Độ trễ phản hồi khi thay đổi khoảng cách là rất thấp (< 500ms).
* Cơ chế xác thực hoạt động hiệu quả, loại bỏ hoàn toàn nhiễu từ các thiết bị Bluetooth lạ xung quanh (điện thoại, tai nghe...).

## 6. KẾT LUẬN & HƯỚNG PHÁT TRIỂN

### 6.1. Kết luận

Đề tài đã hoàn thành việc xây dựng một hệ thống cảnh báo an ninh không dây hoàn chỉnh từ lý thuyết đến thực tiễn.

* **Về phần cứng:** Tận dụng tốt sức mạnh của chip ESP32 (Dual-core, BLE/WiFi tích hợp).
* **Về phần mềm:** Áp dụng thành công kiến trúc đa nhiệm FreeRTOS, giúp tách biệt rạch ròi giữa các tác vụ xử lý sóng (Logic) và tác vụ hiển thị (UI).

### 6.2. Hướng phát triển

* Tích hợp thêm module Relay để điều khiển khóa chốt điện từ (Solenoid Lock) thực tế.
* Sử dụng chế độ Deep Sleep trên Trạm Phát để kéo dài tuổi thọ pin lên đến 6 tháng - 1 năm.
* Phát triển App Mobile để cấu hình Key và xem nhật ký ra vào.