Lập trình cho esp32 kết nối với RFID RC522

**I. Quy trình thực hiện**

**1. Chuẩn bị**

* **Tải và cài đặt thư viện cần thiết**:
  + **Thư viện MFRC522**: Tải thư viện từ **Sketch > Include Library > Manage Libraries**, tìm "MFRC522" của Miguel Balboa, cài đặt phiên bản mới nhất .
  + **Thư viện HTTPClient**: Là thư viện chuẩn của ESP32, không cần cài đặt riêng, nhưng đảm bảo Arduino IDE đã cập nhật.
  + **Thiết bị phần cứng**: ESP32 Dev Module, module RFID RC522, dây nối, nguồn cấp điện, và máy tính có kết nối Internet.

**2. Kết nối và mã hóa**

* **Kết nối phần cứng**:
  + Kết nối module RC522 với ESP32:
    - SDA (SS) → GPIO 21
    - SCK → GPIO 18
    - MOSI → GPIO 23
    - MISO → GPIO 19
    - RST → GPIO 22
    - GND → GND
    - 3.3V → 3.3V
  + Kết nối ESP32 với máy tính qua cổng USB.
* **Mã hóa**:
  + Sử dụng mã Arduino để đọc thẻ RFID và gửi dữ liệu lên Firebase Realtime Database qua HTTPClient. Mã được xây dựng dựa trên yêu cầu trước, với các thông số:
    - WiFi SSID và mật khẩu.
    - Firebase Host: https://esp32-166c4-default-rtdb.firebaseio.com.
    - Firebase API Key: AIzaSyC3tOt-MRumNTeDClxxvhQQ05mLlVhDBjo.

Code chính:

|  |
| --- |
| #include <SPI.h>  #include <MFRC522.h>  #include <WiFi.h>  #include <HTTPClient.h>  #include <time.h>  #define WIFI\_SSID "Kvv"  #define WIFI\_PASSWORD "66668888"  #define FIREBASE\_HOST "https://esp32-166c4-default-rtdb.firebaseio.com"  #define FIREBASE\_API\_KEY "AIzaSyC3tOt-MRumNTeDClxxvhQQ05mLlVhDBjo"  #define SS\_PIN 21  #define RST\_PIN 22  MFRC522 rfid(SS\_PIN, RST\_PIN);  unsigned long lastScanTime = 0;  const unsigned long scanInterval = 2000;  void setup() {    Serial.begin(115200);    SPI.begin();    rfid.PCD\_Init();    Serial.println("RFID RC522 ready.");      WiFi.begin(WIFI\_SSID, WIFI\_PASSWORD);    Serial.print("Connecting to WiFi...");    int wifiTimeout = 0;    while (WiFi.status() != WL\_CONNECTED && wifiTimeout < 30) {      delay(500);      Serial.print(".");      wifiTimeout++;    }    if (WiFi.status() == WL\_CONNECTED) {      Serial.println("Connected to WiFi");      Serial.println("IP Address: " + WiFi.localIP().toString());    } else {      Serial.println("Failed to connect to WiFi");      return;    }      const char\* ntpServers[] = {"pool.ntp.org", "asia.pool.ntp.org", "time.google.com", "time.windows.com"};    bool ntpSynced = false;    for (int i = 0; i < 4; i++) {      configTime(7 \* 3600, 0, ntpServers[i]);      Serial.print("Trying NTP server: ");      Serial.println(ntpServers[i]);      delay(5000);      time\_t now = time(nullptr);      if (now > 100000) {        Serial.println("NTP time synced: " + String(ctime(&now)));        ntpSynced = true;        break;      }    }    if (!ntpSynced) {      Serial.println("Failed to sync NTP time with all servers. Using default timestamp.");    }  }  void loop() {    if (millis() - lastScanTime >= scanInterval) {      if (rfid.PICC\_IsNewCardPresent() && rfid.PICC\_ReadCardSerial()) {        // Đọc ID thẻ        String tagID = "";        for (byte i = 0; i < rfid.uid.size; i++) {          tagID += String(rfid.uid.uidByte[i], HEX);        }        tagID.toUpperCase();        Serial.println("Tag ID: " + tagID);          time\_t now = time(nullptr);        String timeStr;        if (now < 100000) {          timeStr = "2025-07-09T14:40:00";          Serial.println("Using default timestamp");        } else {          char timeBuf[20];          strftime(timeBuf, sizeof(timeBuf), "%Y-%m-%dT%H:%M:%S", localtime(&now));          timeStr = String(timeBuf);        }          String path = "/scans/scan\_" + String(millis()) + ".json?auth=" + FIREBASE\_API\_KEY;        String url = String(FIREBASE\_HOST) + path;        String payload = "{\"tag\_id\":\"" + tagID + "\",\"timestamp\":\"" + timeStr + "\"}";          HTTPClient http;        http.begin(url);        http.addHeader("Content-Type", "application/json");        Serial.print("Sending data to Firebase: " + path);        int httpCode = http.PATCH(payload);        if (httpCode == HTTP\_CODE\_OK) {          Serial.println(" - Success");        } else {          Serial.println(" - Failed, HTTP code: " + String(httpCode));          Serial.println("Response: " + http.getString());        }        http.end();        lastScanTime = millis();        rfid.PICC\_HaltA();      }    }  } |

**3. Test thử trên console feedback**

* **Nạp mã**:
  + Trong Arduino IDE, chọn **Công cụ > Board > ESP32 Arduino > ESP32 Dev Module**, chọn cổng COM, nhấn **Tải lên**.
* **Mở Serial Monitor**:
  + **Công cụ > Serial Monitor**, đặt baud rate là 115200.
* **Kiểm tra kết quả**:
  + Quan sát phản hồi:
    - RFID RC522 ready.: Module RC522 khởi động thành công.
    - Connected to WiFi và IP Address: ...: ESP32 kết nối WiFi thành công.
    - Tag ID: D6FDB8F9: Đọc thành công ID thẻ RFID.
    - Sending data to Firebase: Success: Dữ liệu gửi lên Firebase thành công.

Minh họa kết quả:  
Connected to WiFi

IP Address: 172.20.10.2

Trying NTP server: pool.ntp.org

NTP time synced: Wed Jul 9 14:43:24 2025

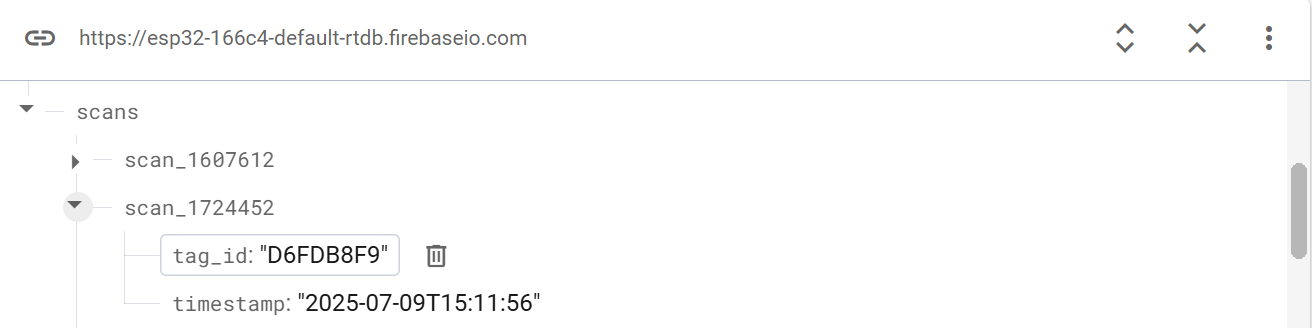
Tag ID: D6FDB8F9

Sending data to Firebase: /scans/scan\_23154.json?auth=AIzaSyC3tOt-MRumNTeDClxxvhQQ05mLlVhDBjo - Success

**II. Mở rộng việc**

**1. Đẩy dữ liệu lên Firebase**

* **Đẩy dữ liệu lên Firebase :**
  + Mã ESP32 sử dụng HTTPClient.h để gửi dữ liệu lên nhánh /scans qua REST API Firebase. Kết quả:



* Kiểm tra trong [Firebase Console](https://console.firebase.google.com/) > **Realtime Database** > **Data**.

**2. Lấy dữ liệu từ Google Spreadsheet**

Sử dụng Google Apps Script để lấy dữ liệu ánh sạ từ firebase ghi vào Google Sheets.

**Cài đặt**:

* Mở Google Sheet mới, đặt tiêu đề: Tag ID, Tên, Thời gian.
* Tạo mã trong Apps Script

|  |
| --- |
| function fetchFirebaseData() {  const databaseUrl = "https://esp32-166c4-default-rtdb.firebaseio.com";  const apiKey = "AIzaSyC3tOt-MRumNTeDClxxvhQQ05mLlVhDBjo";    const tagsResponse = UrlFetchApp.fetch(`${databaseUrl}/rfid\_tags.json?auth=${apiKey}`);  const tagsData = JSON.parse(tagsResponse.getContentText());    const scansResponse = UrlFetchApp.fetch(`${databaseUrl}/scans.json?auth=${apiKey}`);  const scansData = JSON.parse(scansResponse.getContentText());    const sheet = SpreadsheetApp.getActiveSpreadsheet().getActiveSheet();  sheet.clear();  sheet.appendRow(["Tag ID", "Tên", "Thời gian"]);    for (let scanId in scansData) {  const scan = scansData[scanId];  const tagId = scan.tag\_id;  const name = tagsData && tagsData[tagId] ? tagsData[tagId] : "Unknown";  sheet.appendRow([tagId, name, scan.timestamp]);  }  } |

* Quan sát cột Tag ID, Tên, Thời gian được cập nhật tự động mỗi phút nhờ trigger.
* Kiểm tra dữ liệu mới bằng cách quét thẻ RFID, chờ 1 phút, làm mới Google Sheet (nhấn **F5**).

Hình ảnh thực tế:

