Trong các bài viết trước, mình đã hướng dẫn các bạn các phương pháp cập nhật fimrware cho esp qua webserver, internet. Trong bài này, mình sẽ tiếp tục với phương pháp cập nhật thông qua ble

Trước tiên, hãy tải thư viện ble ota:

- Cho arduino: https://github.com/daonguyen207/arduino_ble_ota
- Cho esp-idf: https://github.com/daonguyen207/espidf_ble_ota
- https://app.box.com/s/fplupvveydcq0fnuq13id0994m3u7h5m
- https://app.box.com/s/tqmbvwu0sb3zqi3n5s0ojuvuhrze9x9l

Sử dụng

1. Đối với arduino

Các bạn tham khảo ví dụ mẫu trong phần example

```
#include <BLEDevice.h>
#include <BLEServer.h>
#include <BLEUtils.h>

#include "IOT47_BLE_OTA.h"

#define SERVICE_UUID "55072829-bc9e-4c53-0003-74a6d4c78751"
String bleServerName;

static BLEUUID BLE_UUID(SERVICE_UUID);
BLECharacteristic       ch1_BLECharacteristic(SERVICE_UUID,
BLECharacteristic::PROPERTY_NOTIFY
                                                                  |
BLECharacteristic::PROPERTY_READ
                                                                  |
BLECharacteristic::PROPERTY_WRITE
                                                                  |
BLECharacteristic::PROPERTY_NOTIFY
                                                                  |
BLECharacteristic::PROPERTY_INDICATE);
BLEDescriptor     ch1_Descriptor(BLEUUID((uint16_t)0x2902));

static BLERemoteCharacteristic* ch1_Characteristic;

bool deviceConnected = false;
class MyServerCallbacks: public BLEServerCallbacks {
  void onConnect(BLEServer* pServer) {
    deviceConnected = true;
    Serial.println("Client connect");
```

```
  };
  void onDisconnect(BLEServer* pServer) {
    deviceConnected = false;
    iot47_stop_ota();
    Serial.println("Client disconnect");
    pServer->startAdvertising();
  }
};

class MyCallbacks: public BLECharacteristicCallbacks {
    void onWrite(BLECharacteristic *pCharacteristic) {
      std::string rxValue = pCharacteristic->getValue();
                                        if(iot47_ota_task((uint8_t
*)&(rxValue[0]),rxValue.length()))return; //b□t buộc ph□i gọi □ đây

      //ph□n này x□ lí nhận data ble c□a user
      if (rxValue.length() > 0) {
        Serial.println("ble len: " + String(rxValue.length()));
        Serial.print("Received Value: ");
        for (int i = 0; i < rxValue.length(); i++)
          Serial.print(rxValue[i]);

        Serial.println();
        Serial.println("*********");
      }
    }
};
void setup()
{
   Serial.begin(115200);
  bleServerName = "IOT47 BLE OTA TEST";
  Serial.println("Starting BLE work!");
  BLEDevice::init(bleServerName.c_str());
  BLEDevice::setMTU(512);
  BLEServer *pServer = BLEDevice::createServer();
  pServer->setCallbacks(new MyServerCallbacks());

  //Create the BLE Service
  BLEService *bmeService = pServer->createService(SERVICE_UUID);

  // Create BLE Characteristics and Create a BLE Descriptor
  bmeService->addCharacteristic(&ch1_BLECharacteristic);
  ch1_Descriptor.setValue("BLE chanel 1");
  ch1_BLECharacteristic.addDescriptor(&ch1_Descriptor);

  ch1_BLECharacteristic.setCallbacks(new MyCallbacks());
```

Tài liệu này được tải từ website: http://linhkienthaomay.com. Zalo hỗ trợ: 0389937723

```
  // Start the service
  bmeService->start();

  // Start advertising
  BLEAdvertising *pAdvertising = BLEDevice::getAdvertising();
  pAdvertising->addServiceUUID(SERVICE_UUID);
  pServer->getAdvertising()->start();

  //ch1_Characteristic = bmeService->getCharacteristic(BLE_UUID);
  //ch1_BLECharacteristic->registerForNotify(notifyCallback);
  Serial.println("Waiting a client connection to notify...");


  iot47_ble_ota_begin(&ch1_BLECharacteristic); //b□t buộc ph□i gọi

  //đăng kí callback, tùy user
  iot47_ble_ota_set_begin_callback([](uint32_t curen, uint32_t totol){
      Serial.println("Begin ota");
    });
    iot47_ble_ota_set_proces_callback([](uint32_t  curen,  uint32_t
totol){
    Serial.print(curen);
    Serial.print("/");
    Serial.println(totol);
  });
  iot47_ble_ota_set_end_callback([](uint32_t curen, uint32_t totol){
   Serial.println("Download done");
  });
  iot47_ble_ota_set_error_callback([](uint32_t curen, uint32_t totol){
   Serial.println("Download error");
  });
}

void loop()
{

}
```

2. Đối với esp-idf

Vào sdk config, bật bluetooth

Add thư viện **espidf_ble_ota.h** và **easy_ble.h**

Phiên bản cho esp-idf mình có suport thêm 1 lớp bọc giúp khởi tạo ble nên việc sử dụng còn dễ dàng hơn

easy_ble là lớp bọc khởi tạo bluetooth giúp bạn nhanh chóng sử dụng thư viện chỉ với 3 dòng khởi tạo. Tuy nhiên bạn sẽ ít có quyền can thiệp vào việc khởi tạo ble hơn

```
#include <esp_wifi.h>
#include <esp_event.h>
#include <esp_log.h>
#include <esp_system.h>
#include <nvs_flash.h>
#include <sys/param.h>
#include "nvs_flash.h"
#include "espidf_ble_ota.h"
#include "easy_ble.h"
void app_main(void)
{
    ESP_ERROR_CHECK(nvs_flash_init());
    uint8_t mac[6];
    esp_wifi_get_mac(ESP_IF_WIFI_STA, mac);
    easy_ble_init(mac);

    ble_ota_init();
}
```

Đăng kí các callback

Dưới dây là các callback trong quá trình ota, bạn có thể sử dụng để biết qua trình ota đang diễn ra như nào (Ví dụ in % download ra màn hình LCD hoặc thông báo lỗi ...)

```
typedef void (*ota_callback_t)(uint32_t curen, uint32_t totol);void
iot47_ble_ota_set_begin_callback(ota_callback_t       c);void
iot47_ble_ota_set_proces_callback(ota_callback_t      c);void
iot47_ble_ota_set_end_callback(ota_callback_t       c);void
iot47_ble_ota_set_error_callback(ota_callback_t c);


typedef void (*easy_ble_callback_t)(uint8_t *rxValue, uint8_t
len,esp_gatt_if_t *gatts_if, uint16_t conn_id, uint16_t
attr_handle);void easy_ble_set_rx_callback(easy_ble_callback_t c);
```

Không sử dụng lớp bọc easy_ble

TRUNG TÂM SỬA CHỮA ĐIỆN TỬ QUẢNG BÌNH

MR. XÔ - 0901.679.359 - 80 Võ Thị Sáu, Phường Quảng Thuận, tx Ba Đồn, tỉnh Quảng Bình

Bằng cách sử dụng trực tiếp các api của espidf_ble_ota, bạn có thể toàn quyền khởi tạo ble và sử dụng ble theo ý muốn bạn, tuy nhiên sẽ cần 1 chút kiến thức về ble. Chỉ cần đặt hàm iot47_ble_ota_task vào trong sự kiện ESP_GATTS_WRITE_EVT Ví dụ:

```
case ESP_GATTS_WRITE_EVT: {          if (!param->write.is_prep){
if(iot47_ble_ota_task(param->write.value,param->write.len,&gatts_if,pa
ram->write.conn_id,param->write.handle) == 0)               {
//user code                 ESP_LOGI(GATTS_TAG, "GATT_WRITE_EVT A,
conn_id %d, trans_id %d, handle %d", param->write.conn_id,
param->write.trans_id,                param->write.handle);
ESP_LOGI(GATTS_TAG, "GATT_WRITE_EVT A, value len %d, value :",
param->write.len);                esp_log_buffer_hex(GATTS_TAG,
param->write.value, param->write.len);                }        }
example_write_event_env(gatts_if,  &a_prepare_write_env,  param);
break;     }
```

Hàm này là hàm ưu tiên nên hãy đặt nó ở đầu. Nếu nó trả về khác 0 , tức là quá trình OTA đang diễn ra, do vậy bạn chỉ xử lí các tác vụ ble của user nếu hàm này trả về 0 Và để an toàn hơn, hãy đặt iot47_stop_ota(); vào sự kiện ESP_GATTS_DISCONNECT_EVT. Sau khi khởi tạo ble xong, chỉ cần gọi ble_ota_init();

Các bài viết tương tự:

1. Bảo mật chương trình cho esp8266
2. Cập nhật firmware từ xa cho esp8266/esp32 (OTA) qua web server
3. Chế mạch báo acquy đầy hay yếu đơn giản
4. LG [L177WSB-PSF – hình bị nhòe và mờ
5. mainboard foxconn h61 cpu g1620@ 2.70ghz – máy chạy rất chậm zô net nghe nhạc cũng kg dk
6. máy giặt lồng đứng – mới
7. máy giặt panasonic F70A6 lồng đứng – + máy bật nguồn để khoảng 30s máy tự động kéo xả .nhung khi bật chạy thì lại ngát xả và cấp nuocs giạt bình thường nhung đến lần

giạt thứ 2 thì lại tụ đọng kéo xả và cáp nuocs nhung khi nhấc canh của hoạc án tạm dùng sau đó bám lai thì lại haotj động bình thường

8. may in lase hp1300 – may guồng khởi động xong đèn vàng nhấp nháy .không in tet được.
9. tu lanh – lung giàn lanh
10. Tủ lạnh samsung inventer – Bị đóng tuyết ngăn đá, không mát ngăn mát
11. Tủ Lạnh Sharp điều khiển bằng board. – Mất lạnh, đèn vẫn sáng, quạt vẫn quay, Log chạy một lúc là ngắt, phin lọc nóng hơn bình thường. Thay thermich vẫn không được. Thay 2 con sensor là log không chạy nữa. Dòng cao. Xin các cao thủ giúp đỡ và chỉ giáo.Tiện thể cho e hỏi tìm mua board con này có khó không? Giá bao nhiêu?.
12. Tủ panasonic 280L mạch điện tử – cấp nguồn sấy và lock đêu chạy mình đã thay cảm biến rồi mà không được