

NodeMCU: 03_ESPNOW (Emisor y receptor - primeras pruebas)

 agrportfolioeducativo.blogspot.com/2020/03/nodemcu-03espnow-emisor-y-receptor.html

Receptor

/*

Rui Santos

Complete project details at <https://RandomNerdTutorials.com/esp-now-esp8266-nodemcu-arduino-ide/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files.

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

Modificado por Aurelio gallardo

*/

```
#include <ESP8266WiFi.h>
```

```
#include <espnow.h>
```

```
// Ejemplo de estructura de enviar datos
```

```
// debe coincidir la estructura del receptor
```

```
typedef struct struct_message {
```

```
char a[32];
```

```
int b;
```

```
float c;
```

```
String d;
```

```
bool e;
```

```
} struct_message;
```

```
// Crear un mensaje estructurado llamado myData
```

```
struct_message myData;
```

```
// Función callback se ejecutará cuando se reciban datos
```

```
void OnDataRecv(uint8_t * mac, uint8_t *incomingData, uint8_t len) {
```

```
memcpy(&myData, incomingData, sizeof(myData));
```

```
Serial.print("Bytes received: ");
```

```
Serial.println(len);
```

```
Serial.print("Char: ");
```

```
Serial.println(myData.a);
```

```
Serial.print("Int: ");
```

```
Serial.println(myData.b);
```

```
Serial.print("Float: ");
```

```
Serial.println(myData.c);
```

```
Serial.print("String: ");
```

```
Serial.println(myData.d);
```

```
Serial.print("Bool: ");
```

```
Serial.println(myData.e);
```

```
Serial.println();
```

```
}
```

```
void setup() {
```

```
// Inicializa el monitor serie
```

```
Serial.begin(115200);
```

```
// Establece el dispositivo como estación WIFI
```

```
WiFi.mode(WIFI_STA);
```

```
// Iniciar ESP-NOW
```

```
if (esp_now_init() != 0) {
```

```
Serial.println("Error inicializando ESP-NOW");
```

```
return;
```

```
}
```

```
// Una vez ESPNow se inicializa con éxito, usaremos recv CB para
```

```
// obtener la información del paquete recibido
```

```
esp_now_set_self_role(ESP_NOW_ROLE_SLAVE); // Ahora este es el esclavo
```

```
esp_now_register_recv_cb(OnDataRecv); // función callback
```

```
}
```

```
void loop() {
```

```
}
```

[view raw ESPNOW Receiver.ino](#) hosted with ♥ by [GitHub](#)

=====

Emisor

```
/*
```

```
Rui Santos
```

```
Complete project details at https://RandomNerdTutorials.com/esp-now-esp8266-nodemcu-arduino-ide/
```

```
Permission is hereby granted, free of charge, to any person obtaining a copy  
of this software and associated documentation files.
```

```
The above copyright notice and this permission notice shall be included in all  
copies or substantial portions of the Software.
```

```
Modificado por Aurelio Gallardo.
```

```
*/
```

```
#include <ESP8266WiFi.h>
```

```
#include <espnow.h> // Biblioteca ESPNOW
```

```
// Esta es la dirección del ue va A RECIBIR los datos
```

```
uint8_t broadcastAddress[] = {0xA4, 0xCF, 0x12, 0xDF, 0x5A, 0x6B};
```

```
// Ejemplo de estructura de enviar datos
```

```
// debe coincidir la estructura del receptor
```

```
typedef struct struct_message {
```

```
char a[32];
```

```
int b;
```

```
float c;
```

```
String d;
```

```
bool e;
```

```
} struct_message;
```

```
// Crear un mensaje estructurado llamado myData
```

```
struct_message myData;
```

```
// Función callback cuando datos se envían
```

```
void OnDataSent(uint8_t *mac_addr, uint8_t sendStatus) {
```

```
Serial.print("Status del último paquete enviado: ");
```

```
if (sendStatus == 0){
```

```
Serial.println("Entregado con éxito");
```

```
}
```

```
else{
```

```
Serial.println("Entrega fallida");
```

```
}
```

```
}
```

```
void setup() {
```

```
// Inicializa monitor serie
```

```
Serial.begin(115200);
```

```
// Set device as a Wi-Fi Station
```

```
WiFi.mode(WIFI_STA);
```

```
// Init ESP-NOW
```

```
if (esp_now_init() != 0) {
```

```
Serial.println("Error inicializando ESP-NOW");
```

```
return;
```

```
}
```

```
// Once ESPNow is successfully Init, we will register for Send CB to
```

```
// get the status of Trasnmitted packet
```

```
// Una vez que ESPNOW se inicializa con éxito, usaremos Send CB
```

```
// para obtener el estado del paquete transmitido.
```

```
esp_now_set_self_role(ESP_NOW_ROLE_CONTROLLER); // como controlador
```

```
esp_now_register_send_cb(OnDataSent); // define la función de callback
```

```
// Registra pareja
```

```
// La función esp_now_add_peer acepta los argumentos (en orden):
```

```
// mac address, role, wi-fi channel, key, and key length.
```

```
esp_now_add_peer(broadcastAddress, ESP_NOW_ROLE_SLAVE, 1, NULL, 0);
```

```
}
```

```
void loop() {
```

```
// Set values to send
```

```
strcpy(myData.a, "THIS IS A CHAR");
```

```
myData.b = random(1,20);
```

```
myData.c = 1.2;
```

```
myData.d = "Hello";
```

```
myData.e = false;
```

```
// Envía el mensaje via ESP-NOW
```

```
esp_now_send(broadcastAddress, (uint8_t *) &myData, sizeof(myData));
```

```
delay(500);
```

```
}
```

[view raw](#) [ESPNOW Emisor.ino](#) hosted with ♥ by [GitHub](#)

Al principio pensé que no funcionaban los programas. Pero algo me hizo sospechar de una unidad... Así era. Ambos programas funcionan perfectamente.