

Arduino IoT: Instalación y configuración del dispositivo



Sistemas Embebidos

Francisco Javier Pérez Martínez

22 de abril de 2022



Índice

1. Trabajo a realizar	3
1.1. Función Blink	3
1.2. Conexión a Internet	4
A. Programa función blink	5
B. Programa conexión WiFi	5



1. Trabajo a realizar

Para el presente documento, correspondiente a la práctica 2 de la asignatura, se utilizará un dispositivo embebido, concretamente el Arduino Nano 33 IoT. Instalaremos el IDE para trabajar en un entorno de desarrollo y los controladores necesarios. Una vez instalados, realizaremos una serie de ejercicios propuestos: una función de parpadeo haciendo uso del LED del propio dispositivo y finalmente un programa que nos permita conectarnos a Internet mediante la conectividad inalámbrica de la tarjeta utilizando las librerías correspondientes.

1.1. Función Blink

En primer lugar, cargamos en el entorno el ejemplo de Blink del LED que viene instalado en el IDE.

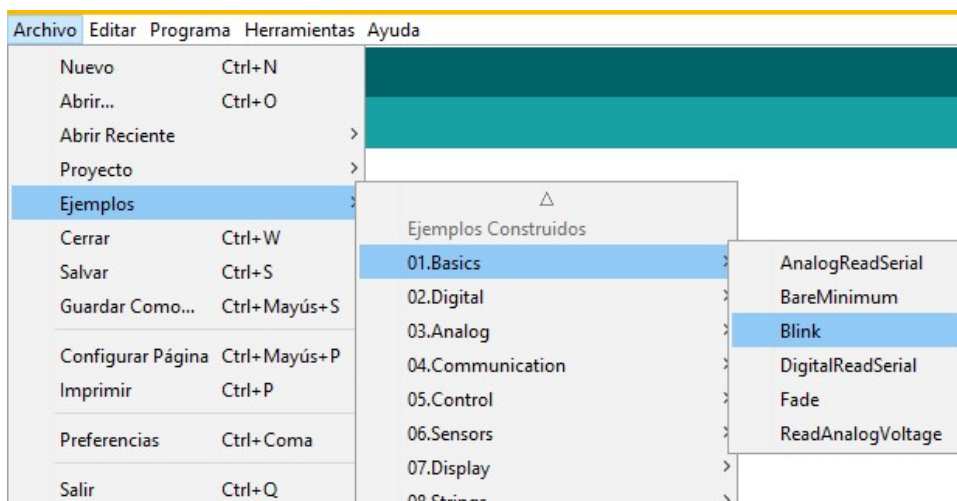


Figura 1: Ejemplo blink

A partir del código de ejemplo, se ha modificado el programa para que aumente la frecuencia de parpadeo a 100ms y que muestre una secuencia de intermitencias consecutivas de 1 a 5 parpadeos en bucle.

El programa resultante se puede observar en el anexo A.



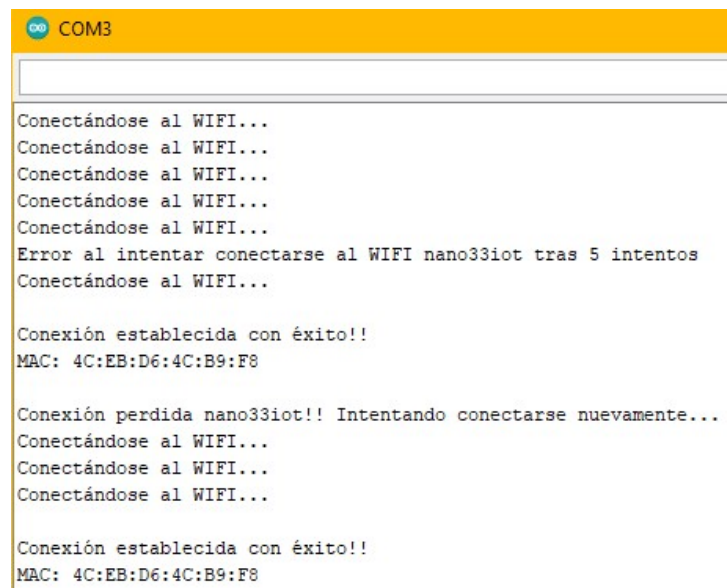
1.2. Conexión a Internet

Para este ejercicio, primero debemos instalarnos la librería de WifiNINA para poder utilizar la conectividad inalámbrica de la tarjeta.

Una vez instalada, configuramos el dispositivo para su conexión a Internet Wifi. Para ello, desde nuestro teléfono móvil se ha creado una zona-wifi para poder conectarse a él. Una vez creado, implementamos lo que se nos pide, si tras 5 segundos el dispositivo no consigue conectarse a Internet se muestra un mensaje de error.

Inicialmente el LED estará encendido. Si no se ha podido establecer conexión, el led parpadeará, y si la conexión se ha establecido correctamente el led se apagará. El programa resultante se puede observar en el anexo B.

Ejemplo de salida del programa:



```
COM3
Conectándose al WIFI...
Conectándose al WIFI...
Conectándose al WIFI...
Conectándose al WIFI...
Conectándose al WIFI...
Error al intentar conectarse al WIFI nano33iot tras 5 intentos
Conectándose al WIFI...

Conexión establecida con éxito!!
MAC: 4C:EB:D6:4C:B9:F8

Conexión perdida nano33iot!! Intentando conectarse nuevamente...
Conectándose al WIFI...
Conectándose al WIFI...
Conectándose al WIFI...

Conexión establecida con éxito!!
MAC: 4C:EB:D6:4C:B9:F8
```

Figura 2: Salida obtenida conexión WiFi

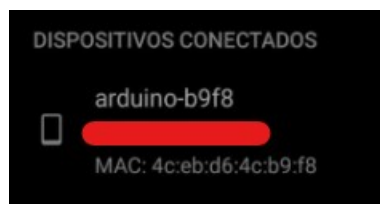


Figura 3: Arduino conectado al teléfono móvil



A. Programa función blink

```
1 // the setup function runs once when you press reset or power the board
2 void setup() {
3     // initialize digital pin LED_BUILTIN as an output.
4     pinMode(LED_BUILTIN, OUTPUT);
5 }
6
7 // the loop function runs over and over again forever
8 void loop() {
9     for (int i = 1; i <= 5; i++) {
10         for (int j = 1; j <= i; j++) {
11             digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
12             delay(100); // frecuencia de parpadeo a 100ms
13             digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
14             delay(100); // frecuencia de parpadeo a 100ms
15         }
16         delay(1000); // espera 1 segundo para la siguiente secuencia 1..5
17     }
18 }
```

B. Programa conexión WiFi

```
1 #include <WiFiNINA.h>
2
3 #define SSID      "nano33iot"
4 #define PASSWORD  "12345678"
5
6 int intentos = 0; // numero de intentos
7 bool estado = false; // controla el estado de conexión
8 byte mac[6];
9
10 void Wifi_ON() {
11     Serial.println("Conectándose al WIFI...");
12     WiFi.begin(SSID, PASSWORD);
13 }
14
15 void blink(int n) { // función de parpadeo
16     digitalWrite(LED_BUILTIN, HIGH); // led on
17     delay(n);
18     digitalWrite(LED_BUILTIN, LOW); // led of
19 }
20
21 void setup() {
22     pinMode(LED_BUILTIN, OUTPUT); // inicializa el pin digital LED_BUILTIN como una salida.
```



```
23     Serial.begin(9600); // inicia la comunicación en serie
24
25     digitalWrite(LED_BUILTIN, HIGH); // led on - inicio
26     Wifi_ON();
27 }
28
29 void printMacAddress(byte mac[]) {
30     for (int i = 5; i >= 0; i--) {
31         if (mac[i] < 16) Serial.print("0");
32         Serial.print(mac[i], HEX);
33         if (i > 0) Serial.print(":");
34     }
35     Serial.println();
36 }
37
38 void loop() {
39
40     // si el arduino no esta conectado al wifi, el led parpadea e intenta conectarse
41     if (WiFi.status() != WL_CONNECTED) {
42         intentos += 1;
43         blink(500); // led blink - conexión incorrecta.
44
45         if(estado) {
46             Serial.println("Conexión perdida " + (String) SSID + "!! Intentando conectarse nuevamente...");
47             estado = false;
48             delay(500); // una vez pasado este retardo intenta conectarse de nuevo
49         }
50         Wifi_ON(); // intentar conectarse nuevamente
51
52         if (intentos == 5) { // si tras 5 intentos no se conecta, mostrar mensaje de error.
53             Serial.println("Error al intentar conectarse al WIFI " + (String) SSID + " tras 5 intentos");
54             estado = false;
55             intentos = 0;
56         }
57     }
58     else {
59         if(!estado) {
60             intentos = 0;
61             estado = true;
62             digitalWrite(LED_BUILTIN, LOW); // led off - conexión correcta
63             Serial.println("\nConexión establecida con éxito!!");
64             WiFi.macAddress(mac);
65             Serial.print("MAC: ");
66             printMacAddress(mac);
67             Serial.println();
68         }
69     }
70 }
```
