

Sistemas Embebidos



Práctica 9

Óscar David Tremiño Guirao Marcos Cerdán Amat
Francisco Joaquín Murcia Gómez Francisco Javier Pérez Martínez

Mayo 2022

Índice

1. Introducción	3
2. Programa desarrollado	4
3. Resultados	5

1. Introducción

Esta práctica consiste en aplicar los conocimientos adquiridos en las prácticas anteriores, sobre la utilización de la pantalla OLED y el sensor MAX30102. Esta vez, ya no requeriremos de la placa Arduino UNO, si no que utilizaremos una placa Arduino NANO 33 IOT ya que dispone de conexión a internet y nos será de utilidad a la hora de enviar los datos.

La primera parte de la práctica, va a consistir en mostrar nuestros nombres en firebase, plataforma para el desarrollo de aplicaciones web. La siguiente parte, será enviar la hora actual a dicha plataforma. Por último, vamos a tomarnos las pulsaciones con el sensor MAX30102 y las enviaremos a la plataforma. Para la realización de esta práctica hemos realizado el mismo esquema de conexiones que en la práctica anterior, pero cambiando la placa:

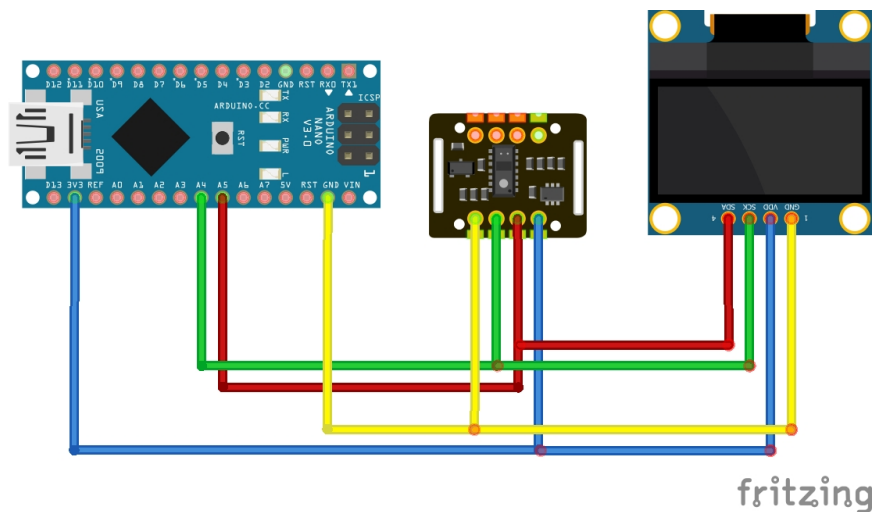


Figura 1: Esquema de conexiones placa arduino nano 33 iot.

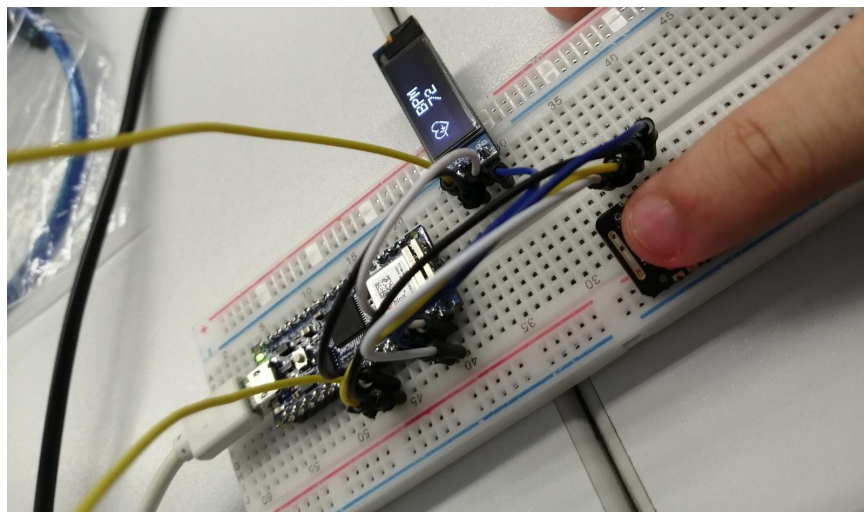


Figura 2: Comprobación del funcionamiento del sistema, MONTAJE.

2. Programa desarrollado

Para realizar el desarrollo de nuestro programa, hemos utilizado el código suministrado en el enunciado y le hemos añadido las funcionalidades siguiendo el ejemplo dado.

Para que nuestro código funcione, primero de todo debemos instalarnos las librerías que nos menciona el ejemplo: "ArduinoHttpClient", "ArduinoJson", "ArduinoECCX08" y "NTPClient". Las otras mencionadas corresponden a librerías WiFi que ya fueron instaladas en prácticas anteriores.

Debemos configurar los siguientes parámetros, ssid y contraseña del punto de acceso de nuestro dispositivo móvil para conectarnos a Internet, definir el cliente NTP y proporcionar el identificador y el id del proyecto en Firebase. Además, debemos indicarle el host donde se encuentra la aplicación.

```
1 // Wifi ssid and password
2 const char* ssid      = "nano33iot";
3 const char* password  = "12345678";
4
5 // Define NTP Client to get time
6 WiFiUDP ntpUDP;
7 NTPClient timeClient(ntpUDP, "pool.ntp.org", 7200);
8
9 // Firebase
10 #define DEVICE_ID "javi-fran-marcos-oscar"
11 #define PROJECT_ID "arduino-iot-se-default-rtdb"
12
13 String host = String(PROJECT_ID) + ".europe-west1.firebaseio.app";
```

Nuestro código quedaría tal que así:

```
1 void loop() {
2     long irValue = particleSensor.getIR();
3     if(irValue >= 7000) { // If a finger is detected
4         fingerDetected();
5
6         if (checkForBeat(irValue) == true) { //If a heart beat is detected
7             showHeartBeatValue();
8             timeClient.update();
9             Serial.println(timeClient.getFormattedTime());
10            if(beatAvg != 0) { // cuando la media no sea 0, enviamos el mensaje.
11                String msg = "{heartRate:" + (String)beatAvg + ", name:'javi-fran-marcos-oscar', time:'"
12                    + (String)timeClient.getFormattedTime() + "'}";
13                DynamicJsonDocument jsonMsg = toJsonDocument(msg);
14                firebaseDatabasePut(DEVICE_ID, jsonMsg); // mediante httpClient enviamos el mensaje en formato
15                    // JSON serializado utilizando la API de firebase
16            }
17        }
18    }
19    else {
20        noFingerDetected();
21    }
22 }
```

3. Resultados

Una vez que la placa arduino se ha conectado al Wifi, cuando la media de frecuencia cardíaca no sea 0, publicará el mensaje en formato JSON a firebase:

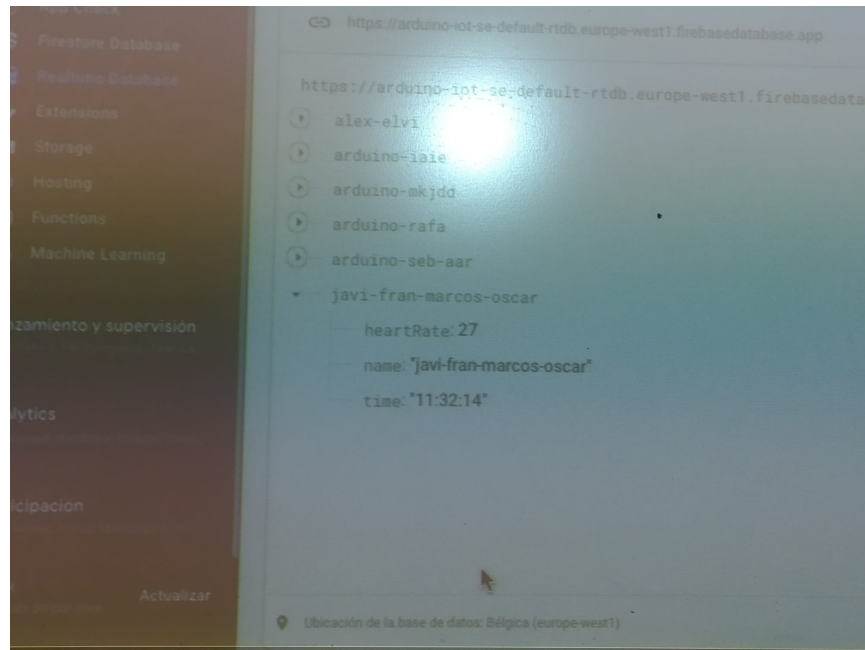


Figura 3: Diversos JSONs enviados por la clase a firebase

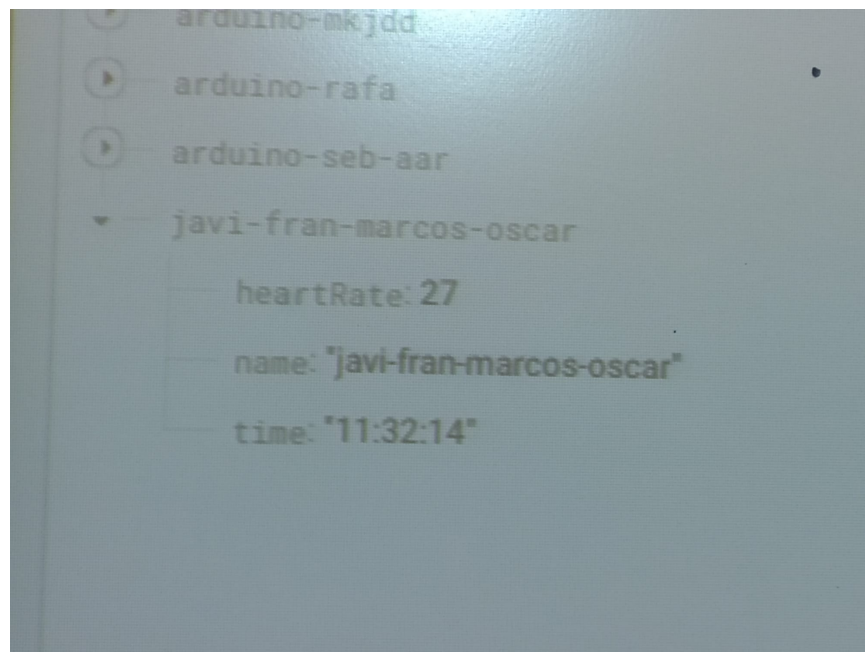


Figura 4: JSON mostrado por firebase