**Dossier de fabrication Radio**

Communication Arduino - Firebase - Android Studio

**Réalisation :**

Batard Jean-Baptiste

Maurize Lucas

Groupe B2 GEII 2ère année

IUT de l’Aisne, Cuffies, 2022

**🚨 Un effort à été fournie dans les commentaires des programmes pour mieux les 🚨 🚨 comprendres donc les explications sur le compte rendu sont faible 🚨**

1. **Présentation et Matériel nécessaire**

I/ Présentation du sujet

II/ Matériel nécessaire

1. **Arduino**

I/ Envoie et Réception de la température et humidité avec le nrf24L01

II/ Générer des valeurs avec une base de temps

III/ Envoyer les données sur la firebase (côté Graphique) (côté données pour Android)

IV/ Programmes Finaux

1. **Firebase**

I/ Comment Firebase Fonctionne

II/ Début d’utilisation de la firebase avec Excel et Arduino

III/ Utilisation ESP32

1. **Android Studio**

I/ Début d’utilisation de la firebase avec Excel et Arduino

II/ Fonctionnement du programme

**1/ Présentation et Matériel nécessaire**

I/ Présentation du sujet

Notre projet commun porte sur la réalisation d’un système de communication radio de données sur une application, permettant le suivi des conditions à l’intérieur d’une pièce et à l’extérieur. Il comprend l’utilisation d’un capteur utilisé 2 fois (DHT22) et un capteur de transmission radio (NRF24L01+). Il se divise en trois parties : une partie sur la firebase, une partie sur arduino et une partie sur android studio.

La partie sur Arduino concerne la récupération des données humidité et température qui sont envoyées bidirectionnellement entre elles pour enfin les afficher avec la transmission radio sur la firebase et aussi la récupération de la variable LED pour simuler une LED.

La partie firebase permet le stockage de données en temps réel et stocke des données en fonction du temps pour permettre de créer un graphique si voulue enfin une variable LED est créée avec 1 = ON et 0= OFF.

La partie Android Studio permet sur une application mobile d’afficher les valeurs températures et humidités en intérieur/extérieur en temps réel avec un simple bouton de déclenchement d’affichage. Avec deux boutons avec un servant à allumer une LED et un autre permettant de l’éteindre.

Cet ensemble permet alors à un utilisateur de connaître en temps réel la température, l’humidité et d’allumer une LED. On peut aussi accéder à toutes ces données sur une base de données avec graphique si voulue.

II/ Matériel nécessaire

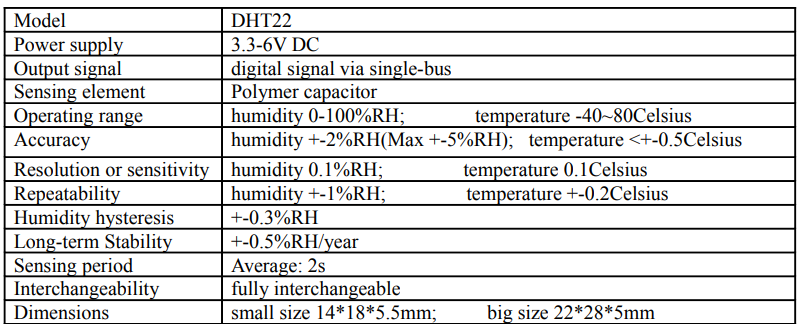
* Cordon d’alimentation USB-C
* Câble micro-USB
* 2 x PC
* 2 x Clavier
* 2 x Souris
* 2 x Écran
* ESP 32
* Arduino Mega
* Logiciel Arduino
* Logiciel Android Studio
* Base de donnée firebase (ici Realtime database)
* Compte google
* Capteurs : NRF24L01+ / DHT22

**2/ Arduino**

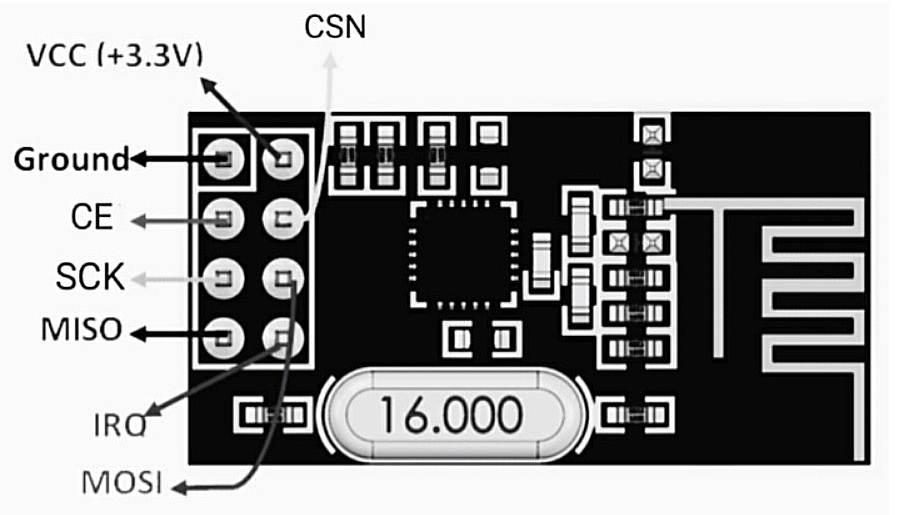
I/ Envoie et Réception de la température et humidité avec le nrf24L01

Présentation des capteurs DHT22 et NRF24L01+ :

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_DHT 22\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ NRF24L01+\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



* It is a transceiver RF module with 2.4GHz.
* The operating voltage is 3.3 Volts.
* The value of nominal current is 50mA.
* The maximum value of the operating current is 250mA.
* It covers a distance of 50-200 feet.
* It uses the SPI communication protocol.
* The range of baud rate is 250kbps to 2Mbps.
* The channel range is 125.
* There are 6 maximum nodes.
* It is a wireless solution available at a low cost.
* It is a single chip GFSK transceiver module with OSI link layer hardware.
* It provides auto ACK, retransmits, address, and CRC computation.
* The data rate on the air is 1Mbps Or 2Mbps.
* The speed of the digital interface (SPI) is 0-8Mbps.
* The range of RF channel operation is 125.
* It provides a short switching time enabling frequency hopping.
* It is completely RF compatible with NRF24XX.
* The tolerance signal input is 5V.
* The power supply range is 1.9V to 3.6V.
* bah je vais mettre que du code avec les commentaires qui explique

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Programme \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Programme de transmission/réception radio de la température et de l’humidité intérieur / extérieur plus transmission de la LED à l’aide du DHT22 et du NRF24L01+ avec un ESP32 :

(Copier coller le code inséré dans l’image)

Programme de transmission/réception radio de la température et de l’humidité intérieur / extérieur plus réception de la LED à l’aide du DHT22 et du NRF24L01+ avec une Arduino Mega :

(Copier coller le code inséré dans l’image)

II/ Générer des valeurs avec une base de temps

A l’aide de la librairie time.h et un serveur qui contient une base de temps en temps réel convertit en UTC +1 (heure de paris), on obtient la base de temps pour effectuer un futur graphique avec les données daté :

(Copier coller le code inséré dans l’image)

III/ Envoyer les données sur la firebase (côté Graphique) (côté données pour Android)

Une fois les données récupéré à l’aide d’un tableau, il ne reste plus qu'à les envoyés a la firebase pour récupérer la température et l’humidité intérieur/extérieur en temps réel ici : 

(Copier coller le code inséré dans l’image)

On récupère aussi les données du tableau pour effectuer un graphique si voulue sur la base de donnée firebase à l’aide de la base de temps on convertit le tout en fichier json pour pouvoir lire les données sur la base de donnée firebase :

(Copier coller le code inséré dans l’image)

IV/ Programme Finale

Programme final ESP32 :

(Copier coller le code inséré dans l’image)

Programme final Arduino Mega :

(Copier coller le code inséré dans l’image)

**3/ Firebase**

I/ Comment Firebase Fonctionne

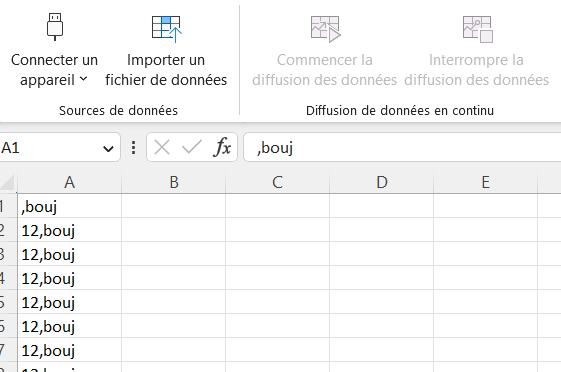
Firebase est une plateforme de développement d'applications mobiles et web proposée par Google. Elle offre une gamme de services, dont une base de données NoSQL en temps réel, qui permet de stocker et de synchroniser les données en temps réel entre différents utilisateurs et appareils, tels que des applications mobiles, des sites web ou des serveurs.

L'utilité de Firebase réside dans sa simplicité et sa facilité d'utilisation. Elle permet aux développeurs de créer rapidement et facilement des applications en temps réel avec des fonctionnalités de synchronisation des données en temps réel.

En ce qui concerne le fonctionnement de la base de données NoSQL de Firebase, elle est basée sur un modèle de données JSON (JavaScript Object Notation) qui permet de stocker les données sous forme de paires clé-valeur. Les données sont organisées en collections, qui sont similaires à des tables dans une base de données relationnelle, et les documents sont les objets qui sont stockés dans ces collections. Chaque document a un identifiant unique, appelé ID de document, et peut contenir des données sous forme de paires clé-valeur.

II/ Début d’utilisation de la firebase avec Excel et Arduino

Au début de notre projet nous a été donné 2 carte arduino nous empêchant donc de créer une connexion directe avec un module Wifi et la fireBase pour cela il faut connecter l’arduino constamment à un Pc avec un logiciel permettant la récupération des données (Exel,CoolTerm..)



et de réussir à convertir ses données en Json ou de les passer sous un fichier google sheet via un site internet ou pour le json un script python qui serait toujours actif sur le PC.

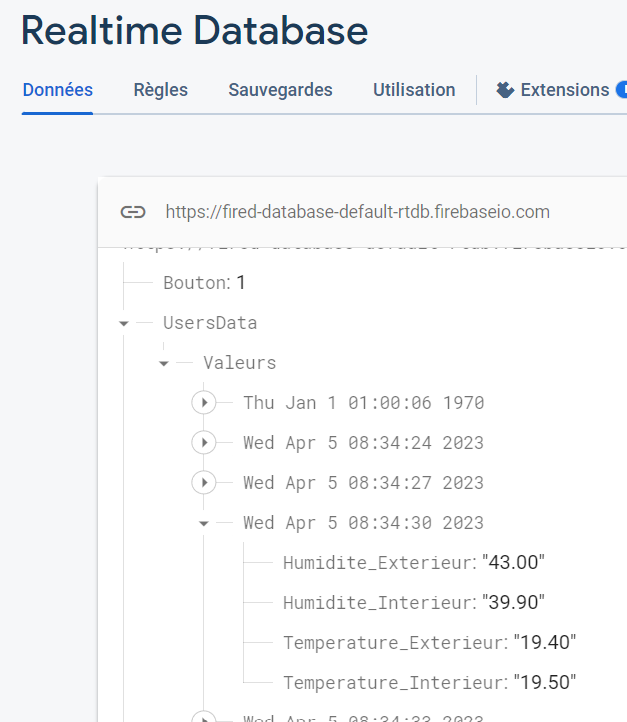
Par la suite, la récupération des données sur firebase est composée de l’id voulu du message et de son chemin d’enregistrement.

III/ Utilisation ESP32

Lors de l'utilisation de l’ESP32, le problème de transmission ne se pose plus les seul problème étant : comment faire une database avec les données en continue dans le temps, comment envoyer ses données .

Pour le problème de données dans le temps on utilisera la base de temps faite précédemment et on l’utilisera comme chemin ainsi le chemin de la donnée sera toujour nouveau et on pourra le retrouver :



et le résultat est 

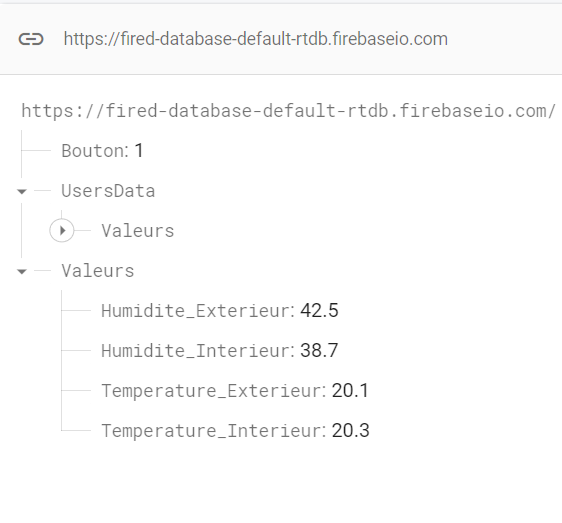
On peut observer l’utilisation des fichiers JSON pour le transfert principal des données. La dernière ligne du programme permet l’envoi des données misent sous forme Json avec le chemin et l’ID déterminé par le chemin de la DataBase

**4/ Android Studio**

I/ Début d’utilisation de la firebase et récupération des données

Pour la récupération des données vers le programme android studio il faut un chemin vers la données stable permettant donc sa collecte or car l’envoie des données comprend un chemin avec une horaire et une date changeante 2 possibilitées s’offre à nous :

on peut utilisateur les données d’un deuxième envoie pour stabiliser le chemin ainsi :



ou bien on peut avec l’utilisation de la base de temps ,envoyer celle-ci en donnée sur FireBase et la réutiliser dans le chemin pour trouver la donnée voulu ainsi on peut avec les différents temps choisir la température et humidité de la date et l’heure voulu.

II/ Fonctionnement du programme

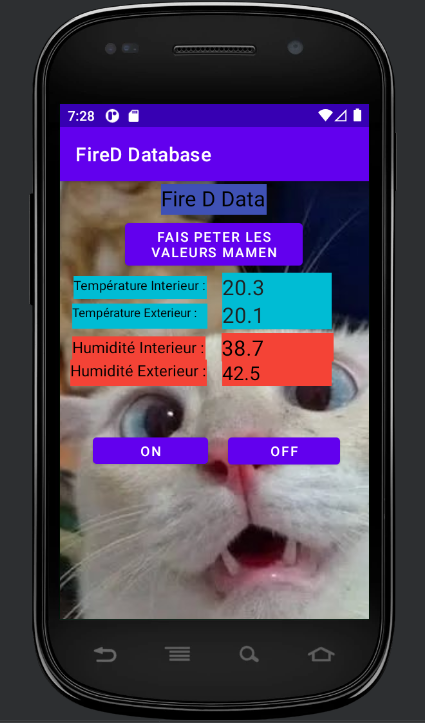
Depuis la firebase nous récupérons la température et l’humidité intérieur / extérieur que nous affichons et nous envoyons la valeur de la LED a la base de donnée realtime data base depuis l’application mobile :

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Programme \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Copier coller le code inséré dans l’image)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Fichier String \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Affichage \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



Pour lier ceci a la base de donnée realtime database, un onglet “Tools” en haut de la toolbar avec un assistant pour la firebase est fourni pour lier directement la base de donnée à android studio et explique clairement et facilement comment lier les deux sans soucis (réalisé ici même).