

## RAPPORT DE PROJET : Développement d'un système de transmission de données audio entre une carte ESP32 et un serveur audio

Ce projet a pour objectif de développer un système de transmission de données audio entre une carte ESP32 et un serveur audio, en utilisant le protocole TCP/IP comme protocole de communication. Le système a été conçu pour surmonter les difficultés liées à la transmission de données audio en temps réel, notamment la gestion des connexions avec le modem et la fragmentation des fichiers.

### **Introduction**

Le projet a consisté à développer un système de transmission de données audio entre une carte ESP32 et un serveur audio, en utilisant le protocole TCP/IP comme protocole de communication. Le système a été conçu pour répondre aux besoins spécifiques de la transmission de données audio en temps réel, notamment la gestion des connexions avec le modem (ou Wi-Fi) et la fragmentation des fichiers.

### Méthodologie

#### **Le projet a été réalisé en plusieurs étapes**

- Développement d'un programme de capture audio pour la carte ESP32, permettant la capture des données audio en temps réel.
- Développement d'un programme de fragmentation des fichiers, permettant de diviser les fichiers audio en morceaux plus petits pour faciliter la transmission.
- Développement d'un programme de transmission des données audio, utilisant le protocole TCP/IP pour transmettre les morceaux de fichiers audio entre la carte ESP32 et le serveur audio.
- Développement d'un programme de réception et de reconstitution des fichiers audio, permettant de reconstituer les fichiers audio à partir des morceaux de fichiers transmis.

## **1-Capture du son et enregistrement en fichier WAV**

L'écriture des données analogiques au format d'un fichier .wav lisible sur toutes les plateformes (Windows, Linux, Android) a nécessité le développement d'un programme spécifique, car l'environnement de développement de l'ESP32 ne propose aucune bibliothèque permettant d'y parvenir. De plus, les ressources disponibles en ligne n'étant pas suffisantes pour remplir cette tâche. Dans le processus de réalisation de ce programme, nous avons suivi les étapes suivantes :

- Définition des paramètres d'entête du fichier : Cette étape indispensable permet de définir la structure du fichier, le débit de lecture, le débit d'écriture, le nombre de canaux et d'autres paramètres qui permettront plus tard à ce fichier d'être lu.
- Récupération du son par I2S et écriture des échantillons Le son est récupéré par le protocole I2S du dispositif INP441 (microphone) et les échantillons sont écrits dans le fichier selon les paramètres d'en-tête du fichier.
- Les 2 étapes ci-dessus succinctement complétées, le son peut être enregistré sur la carte mémoire.

La taille moyenne des fichiers après enregistrement est de 2.5mb.

## **2. Envoie du fichier par morceau.**

Les environnements de programmation de l'ESP32 n'implémentent pas pleinement toutes les fonctionnalités des protocoles de communication tels que HTTP et FTP, qui sont les protocoles généralement utilisés pour le transfert de fichiers. En l'occurrence, les implémentations disponibles ne permettent pas de transférer des fichiers lourds (poids > 1 Mo), ceci en raison des contraintes liées à la taille de la mémoire RAM de la carte. Nous avons donc été contraints de développer un programme qui permet d'envoyer les fichiers par fragments et un autre programme permettant de les reconstituer à leur arrivée sur le serveur.

## **3. Transmission des données par TCP/IP**

Le protocole TCP/IP est un protocole de communication réseau bas niveau permettant d'établir une communication directe entre un client et un serveur. Il offre la possibilité d'avoir le contrôle sur la taille des paquets envoyés et l'intégrité de ces derniers à la

réception, d'où le choix de ce protocole. Nous pouvons donc envoyer les fragments de fichier par paquets de 4 Ko. Ainsi, nous garantissons une utilisation optimale de la carte et nous nous assurons que même les connectivités faibles et instables nous permettront d'envoyer nos fichiers efficacement. Le temps moyen de transmission d'un enregistrement audio varie entre 40 secondes et une minute.

#### 4.Reconstitution des fragments reçu du côté du serveur.

Du côté serveur est déployé un programme permettant de reconstituer les paquets reçus et organise également l'enregistrement des fichiers dans des dossiers correspondant au jour de la réception.

#### 5.Difficultés rencontrées :

Au cours du projet, nous avons rencontré plusieurs difficultés :

- **Gestion des connexions avec le modem** : Complexe et requiert davantage de temps pour palier au divers inconvénients qu'il soulève.
- **Gestion des erreurs de transmission** : La totalité des erreurs de la transmission n'ont pas été gérées en l'occurrence celle lies au perturbation du réseau internet.

#### 6.Résultats

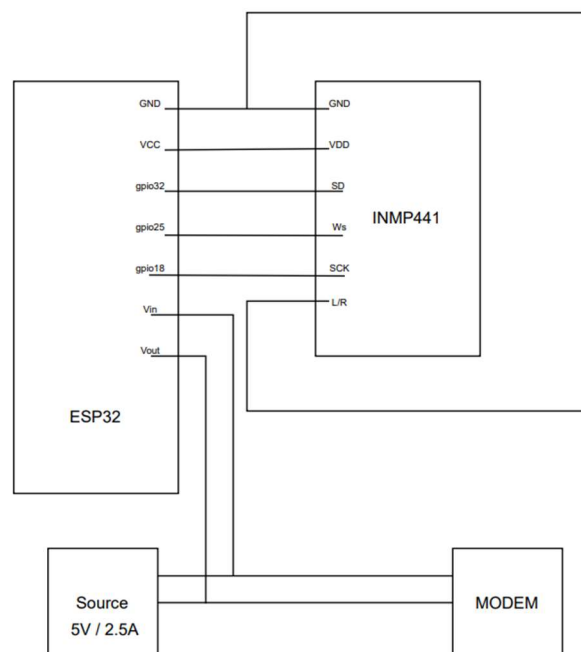
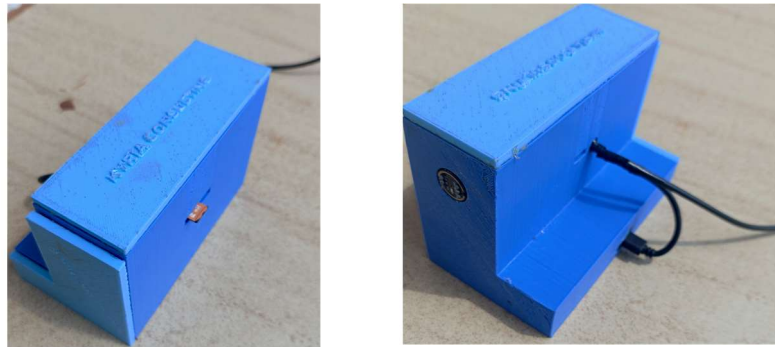


Figure 1 :schéma de connexion électronique du système



*Figure 2 : Montage et assemblage*

- **Temps d'envoi** : Le système envoie les données en 3 min avec une connexion de 6mbps.
- **Taille du fichier après enregistrement** : 2.5 mégabits
- **Ancrage temporel** : Le système envoie les données sur une période de 5 min avec une référence horaire de 00h00. Durant cet intervalle de 5min, le système procède à un enregistrement de 30 secondes, effectue une sauvegarde sur la carte mémoire et envoie le fichier et supprime le fichier est correctement envoyé, sinon le fichier est conservé.
- **Gestion des connexions par le serveur** : Le serveur ne gère qu'une seule connexion à la fois.
- **Comportement en cas d'erreur** : Le système avorte toute transmission et commence un nouvel enregistrement au prochain ancrage temporel
- **Comportement en cas d'écoulement de la période d'envoi** : : Le système avorte toute transmission et commence un nouvel enregistrement au prochain ancrage temporel.
- **Comportement en cas de perte connexion WIFI** : Le système avorte toute transmission tente de se reconnecter jusqu'à ce qu'un wifi soit à nouveau disponible et commence un nouvel enregistrement au prochain ancrage temporel mais sauvegarde les audio enregistrés au différent ancrage durant la période d'indisponibilité.

- **Comportement en cas de rupture de connexion internet :** Le système avorte toute transmission jusqu'à ce que la connexion soit de nouveau disponible, mais sauvegarde les audio enregistrées au différent ancrage durant la période d'inactivité.