a) Logiciels et outils utilisés - MobaXterm : Est un logiciel de gestion de connexions réseau,plus précisément un terminal et aussi un client SSH,il permet de se connecter à des serveurs distants de maniéré très sécurisée.Grace à ma clé privée SSH , j'ai pu accéder au serveur d'Etna et de consulter l' arborescence des projets .Cependant je n'ai pas beaucoup utilisé ce logiciel parce qu'on m'a surtout demandé de me familiariser avec lui et d'avoir les accès nécessaires.Cela était important , car j'étais chargée du développement des programmes liés à la connexion au serveur que je devais ensuite déposer sur ce dernier. - **Arduino IDE :** Est un logiciel de développement utilisé pour écrire , compiler et exécuter du code à base des langages C et C++ et puis le transférer vers des cartes électroniques telles que les ESP8266 ou ESP32 par le port USB .Je l'ai utilisé principalement en codant avec le langage C car il était indispensable dans toutes les tâches qui m'ont été confiées tout au long du stage.

- **CoolTerm :** Est un logiciel de communication série qui permet de se connecter à un port COM via l'USB ou afin d'accéder au port UART .Il est généralement utilisé pour l'envoi et la réception de données, ou pour interagir avec des appareils électroniques. Durant mon stage,il a été utilisé uniquement avec des cartes ESP32-S3 Zero, aussi bien pour l'envoi de trames que pour la vérification des données reçues via la liaison série.

b) Cartes électroniques

- **ESP8266 :** Est une puce Wi-Fi intégrée dan un microcontrôleur qui permet de connecter des objets électroniques à internet ou à un réseau local, afin de les contrôler ou d'échanger des données.C'est la première carte que j'ai utilisée , pour écrire de petits codes dans le but de découvrir le domaine du développement embarqué et m'initier aux bases de la programmation sur microcontrôleurs .

•

- **ESP32-S3-Zero :** Est une puce Wi-Fi et Bluetooth intégrée dans un microcontrôleur ,permettant de connecter des objets électroniques à internet tout en offrant un traitement des données plus rapide et plus sécurisé.C'est la carte sur laquelle j'ai principalement travaillé tout au long de mon stage,où j'ai développé des programmes plus complexes et communiqué avec d'autres appareils.

- **ESP32-P4 nano :** Est une carte de développement, conçue pour les applications embarquées avancées, contrairement aux autres cartes ESP32,elle ne dispose pas de connectivité WiFi ni Bluetooth intégrée,elle permet un traitement plus rapide des données, tout en offrant une sécurité renforcée.C'est la dernière carte que j'ai utilisée , j'y ai importé et testé tous les programmes que j'avais développés auparavant sur l'ESP32-S3 Zero. Dans le but de vérifier et de tester si les programmes de sécurité et de communication étaient compatibles avec la P4, bien qu'elle ne dispose pas de connectivité WiFi ni de Bluetooth.

1.3 Caractéristiques matérielles des microcontrôleurs ESP

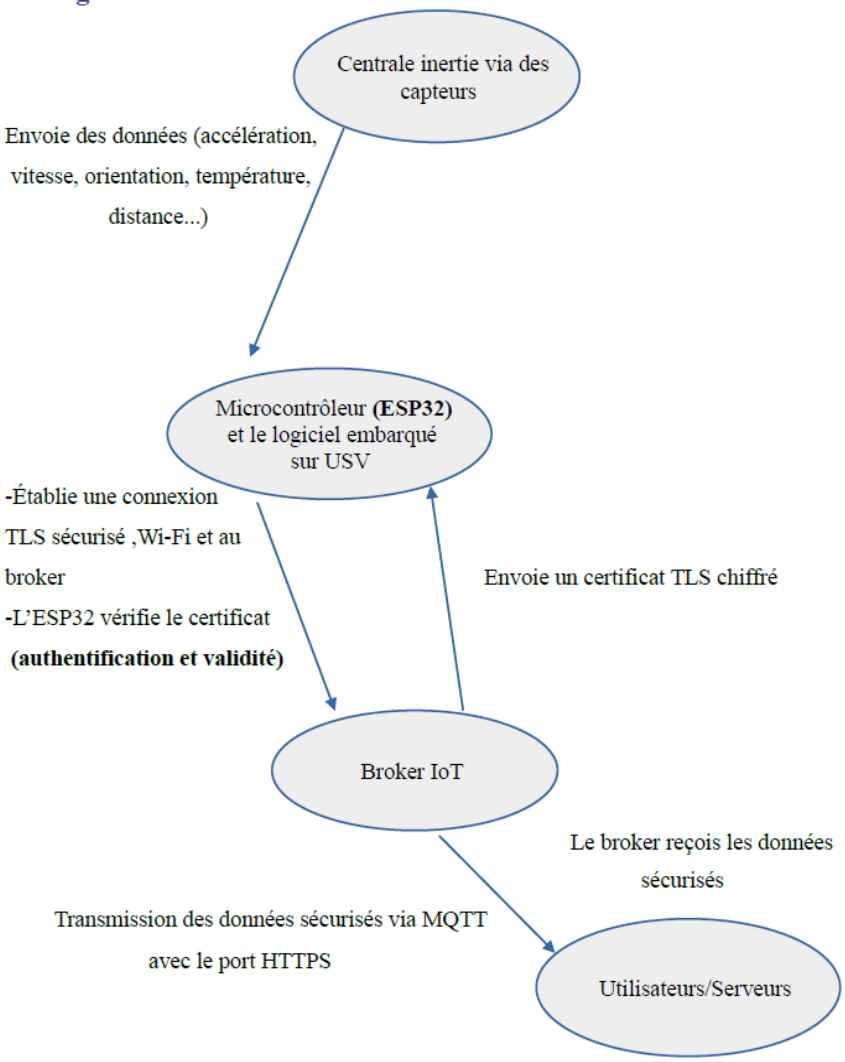
b) ESP32-S3 Zero

Caractéristique	Description
Processeur	Architecture 32 bits
Fréquence CPU	Jusqu'à 240Mhz
RAM	512 Ko SRAM
Mémoire Flash	16Mo pour stocker les programmes
Mode de connexion	Elle fonctionne en mode station ou en point d'accès
Wi-Fi	Intégré avec cryptographie et boot sécurisé
Bluetooth	Intégré
IA/accélérateur	IA intégrée et accélérateur vectoriel efficace pour l'exécution simple
GPIOs	Jusqu'à 44 GPIOs

c) ESP32-P4 nano

Caractéristique	Description
Processeur	Architecture 32 bits
Fréquence CPU	Jusqu'à 400Mhz (HP) et 40MHz(LP)
RAM	768 Ko SRAM + 32 Ko LP SRAM + 32 Mo de PSRAM
Mémoire Flash	16Mo pour stocker les programmes et les données
Mode de connexion	Mode station et point d'accès via module ESP32-C6 intégré
Wi-Fi	Intégré avec cryptographie avancée et boot sécurisé
Bluetooth	Intégré via l'ESP32-C6
IA/accélérateur	IA intégrée et accélérateur très avancé pour les calculs complexes
GIPOs	Jusqu'à 55 GPIOs

2- Description conceptuelle des flux de communication et des échanges chiffrés



3- Description approfondie des missions réalisées

3.1 Mise en place d'une communication sécurisée entre le microcontrôleur et un serveur distant via TLS

J'ai établie une communication sécurisée entre un microcontrôleur de type ESP32 et un serveur distant, avec l'utilisation du Protocole TLS qui permet l'authenticité des données partagées, surtout qu'elles sont partagées via un réseau public.
Ces étapes se résument :

1/Établissement de la connexion au réseau Wi-Fi
Le microcontrôleur doit se connecter au réseau wifi local avec un SSID et un mot de passe donnés par l'entreprise.Le système tente une ou plusieurs tentatives en cas d'échec.

2/Établissement de la connexion TLS
Une fois connecté au réseau Wi-Fi, le microcontrôleur essaie une ou plusieurs fois de se connecter au serveur distant via le Protocole TLS pour une connexion chiffrée grâce au partage de clés de sécurité.

3/Partage de données:
Après les deux connexion Wi-Fi/TLS, il envoie un message au serveur qui peut être des données physiques ,puis il lit la réponse du serveur et l'affiche.

Le programme gère les temps d'attentes , les délais d'échecs de connexion et il répète le processus de connexion en continue afin de garantir la stabilité du réseau. Il utilise aussi une machine d'états pour éviter les bugs et pour reprendre facilement la connexion en cas d'erreurs.

3.2 Développement d'un système de réception et de décodage de trames binaires en temps réel

J'ai développé un programme qui permet de recevoir et de décoder des trames binaires de 25 octets, qui sont envoyées par le logiciel CoolTerm via une liaison série.
Ces étapes se résument :

1/Réception des trames via l'interface série
Le microcontrôleur surveille et d'éclanche un timer juste après la réception du premier octet de la trame.

2/Vérification et validation des trames reçues
À chaque réception d'une trame ,le programme vérifie la longueur de la trame.Dans le cas où elle est égale aux 25 octets, il passe à la vérification de l'octet de start et l'octet de stop (il faut que les deux soient respectivement 0F pour le start et 00 pour le stop).Si elle est bien complète et que les deux octets de start et de stop sont vérifier la trame sera donc bonne et validée.
Une trame est considérée invalide dans les cas suivants :
-Elle dépasse les 25 octets (dépassement de capacité) : dans ce cas, le programme interrompt la lecture, vide le buffer pour éviter la sauvegarde des données, et ignore la

trame. Si cela se produit trois fois consécutivement, le programme affiche un message d'erreur.
- L'un des deux octets (début ou fin) n'est pas correct.
Elle est incomplète,c'est à dire qu'elle contient moins d'octets que la taille exigée. Donc elle ne sera pas traitée.
- Une fois que la trame reçue est vérifiée et validée , le programme les convertit.

3/Affichage des données
C'est la dernière étape ou les valeurs s'affichent sur le serial moniteur et elles sont envoyées au serveur pour la sauvegarde.

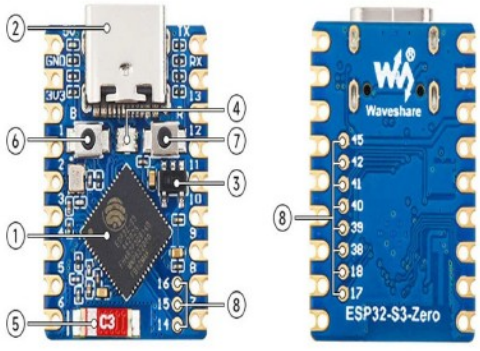
Le programme garde uniquement la trame qui est correcte , qui contient exactement 25 octets et qui est validée. À part cela, il supprime et ignore les autres octets et le vidage du buffer se fait âpres chaque envoie.Le programme utilise aussi une machine a états pour éviter les erreurs et les bugs lors de la réception d'une trame incomplète ou qui dépasse la taille ,cela garantit la stabilité dans le cas de la perte du flux de données.
Tous les programmes et ainsi que les fonctionnalités développées ont été intégrées et testées sur la carte ESP32-P4 Nano. Il était nécessaire de valider la stabilité de la connexion Wi-Fi et TLS sur cette plateforme, ainsi que la bonne réception des trames en conditions réelles car les deux cartes esp32 seront intégrées par la suite dans le drone.

1.2 Détails des cartes électroniques utilisées

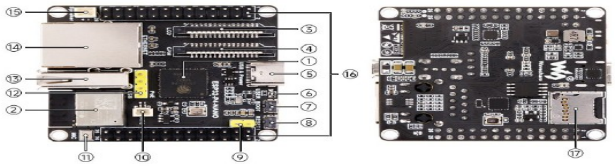
Dans cette partie je vous présenterai les principaux composants des cartes ESP32

a) ESP32-S3 Zero

N°	Composant
1	Microcontrôleur (Wi-Fi,Bluetooth)
2	Connecteur USB Type-C
3	Led d'alimentation
4	Puce de conversion USB vers UART
5	Sélecteur de tension
6	Bouton Reset
7	Bouton Boot
8	Connecteurs d'entrées et sorties



b) ESP32-P4 Nano



N°	Composant
1	Microcontrôleur sans Wi-Fi/Bluetooth
2	Mémoire de stockage
3	Connecteur GPIO 1 d'entrées/sorties
4	Connecteur GPIO 2
5	Connecteur USB Type-C
6	Bouton Reset
7	Bouton Boot
8	LED d'alimentation
9	LED d'utilisateur
10	Régulateur de tension
11	Connecteur d'alimentation externe
12	Connecteur RJ45(Ethernet)
13	Isolation Ethernet
14	Port USB Host Type-A
15	Connecteur UART
16	Trous de montage
17	Lecteur de carte MicroSD

II. Glossaire

-TLS : Transport Layer Security, c'est un protocole de sécurité qui chiffre les communications entre deux appareils .Il garantit l'authenticité et la confidentialité des données partagées.

-USB : Universal Sérial Bus, utilisé pour connecter des périphériques à un ordinateur ou à un microcontrôleur, qui permet la transmission des données.

-UART : Universal Asynchronous Receiver Transmitte, protocole de communication série qui permet les transmissions de données entre un microcontrôleur et les autres périphériques.

-Broker : Un serveur local qui reçoit et distribue les messages et les données entre les appareils.-MQTT : Message Queuing Telemetry Transport, protocole de messagerie basé sur le modèle Publish/ Subscribe, souvent utilisé pour les connexions instables.

-GPIO : General Purpose Input/Output, des broches programmables d'un microcontrôleur, utilisées pour les signaux d'entrées et de sorties.

-Centrale d'inertie : Un capteur qui mesure les mouvements dans l'espace, il est très utilisé dans les robots et les drones.

-Timer : Une fonction permettant de mesurer et de gérer le temps,il est souvent utilisé pour temporiser des actions.

-Délai : Temps d' attente programmé permettant aux actions d'exécuter en prenant le temps nécessaire.

-USV : Unité de sauvegarde d’énergie, un dispositif qui distribue l’alimentation aux équipements électroniques en cas de coupure de courant.

-Connecteur RJ45 : Un connecteur standard utilisé pour les câbles Ethernet ,pour connecter les composants au réseaux LAN.

-Isolation Ethernet : Technique utilisée pour la protection des circuits électroniques en cas de parasites qui viennent des câbles.