

RAPPORT DE PROJET

CARTE DE DEVELOPPEMENT ESP-32

Tao LOPEZ

Jordan CHEVALIER

Mourad HADJI

Table des matières

Objectifs	3
Contraintes Client	3
Outils Utilisés.....	3
Analyse Fonctionnelle niveau 1	4
Explications Schéma	4
Analyse Fonctionnelle niveau 2	5
Explications Schéma	5
Problèmes prévus	6
Schématiques.....	7
Routage	12
Problème rencontré 1	13
Présentation du problème	13
Solution.....	14
Problème rencontré 2	14
Présentation du problème	14
Solution.....	15
Conclusion Projet.....	15
Figure 1 : Connecteurs USB BT et HT	7
Figure 2 : Conversion des tensions en 3V3.....	8
Figure 3 : Portes logiques choix Tension	8
Figure 4 : Circuit logique Falstad	9
Figure 5 : Inversion tension et VDD3 Final.....	9
Figure 6 : USB vers UART	9
Figure 7 : Alimentation ESP32.....	10
Figure 8 : Protocole ESP32	10
Figure 9 : Reset ESP32	10
Figure 10 : IHM Hardware	10
Figure 11 : Mémoire flash	11
Figure 12 : Antenne + Filtrage	11
Figure 13 : Capteurs.....	11
Figure 14 : Connecteurs.....	12

Objectifs

Créer une carte de développement complète comprenant :

- Modules de communication (I2C – SPI – Antenne + filtrage)
- 3 modes d'alimentation (Basse tension – Haute tension – USB-C)
- Capteurs (température – CTN)
- IHM (Led – Buzzer – Bouton RESET)
- Divers (ESD – Connecteurs GROVE – Mikrobus)

Contraintes Client

Général :

1. Avoir la schématique en **mils**.
2. Avoir le PCB en **mm**.
3. Titre, révision et nom du dessinateur attendus sur la schématique

Routage :

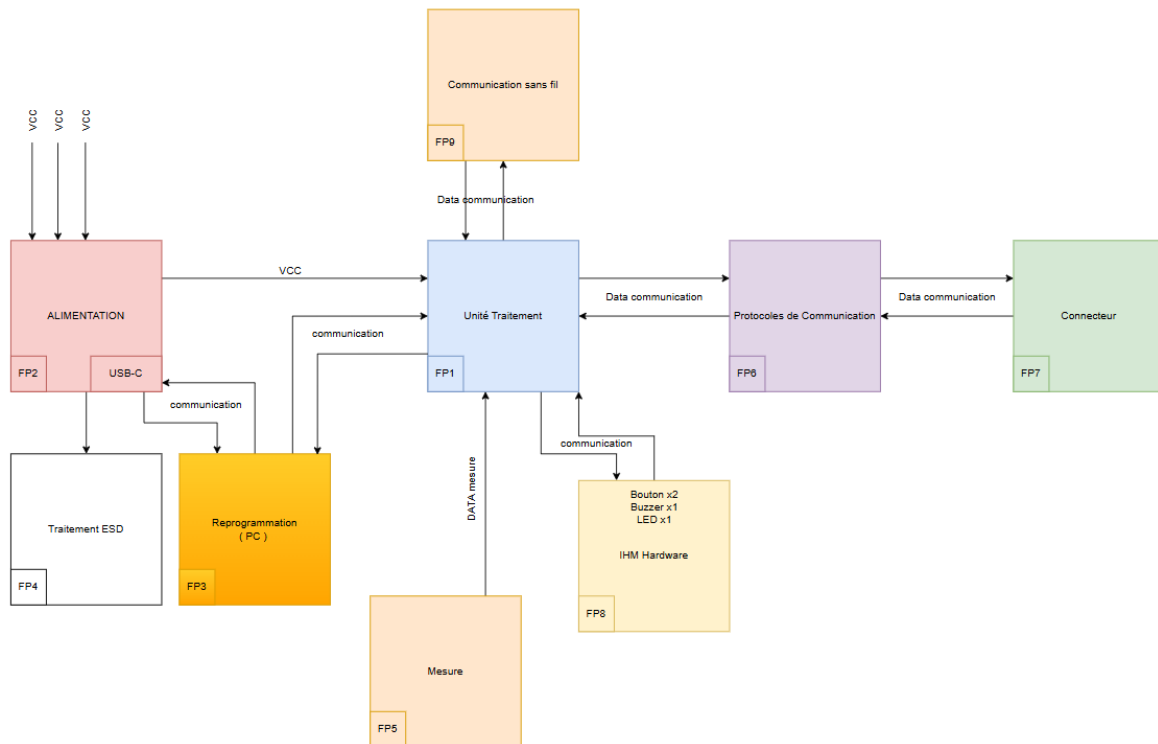
1. Ne pas avoir d'angle droit :
2. Arriver le plus possible au milieu des pads et éviter les raccordements complexifiant les soudures.
3. Utilisation d'un plan de masse obligatoire (et si possible plan d'alimentation)
4. Aucun composant ne doit être mis sur le Bottom
5. Taille des via : pastille de **0.85mm** et trou de **0.35mm**
6. Taille de la sérigraphie (largeur) : **0.15mm**
7. Impédance de 50 Ohms sur la ligne de reprogrammation UART
8. Adaptation du plan de masse ou puissance avec le placement de l'antenne BLE
9. Utilisation des vias d'assemblages sur la couche TOP en fonction des besoins

Outils Utilisés

Les outils utilisés sont :

- Draw.io pour les schémas fonctionnels
- Github
- Altium
- Teams

Analyse Fonctionnelle niveau 1



Explications Schéma

L'analyse fonctionnelle du système vise à définir les fonctionnalités et contraintes de la carte électronique.

1. Alimentation : La carte doit gérer trois sources d'alimentation avec priorité définie : USB (5V, priorité 0), basse tension (2,7-10V, priorité 1) et haute tension (jusqu'à 36V, priorité 2), et commuter automatiquement entre elles.

2. Mesure : Intégration d'un capteur INA pour mesurer courant et température afin de surveiller l'énergie et prévenir la surchauffe.

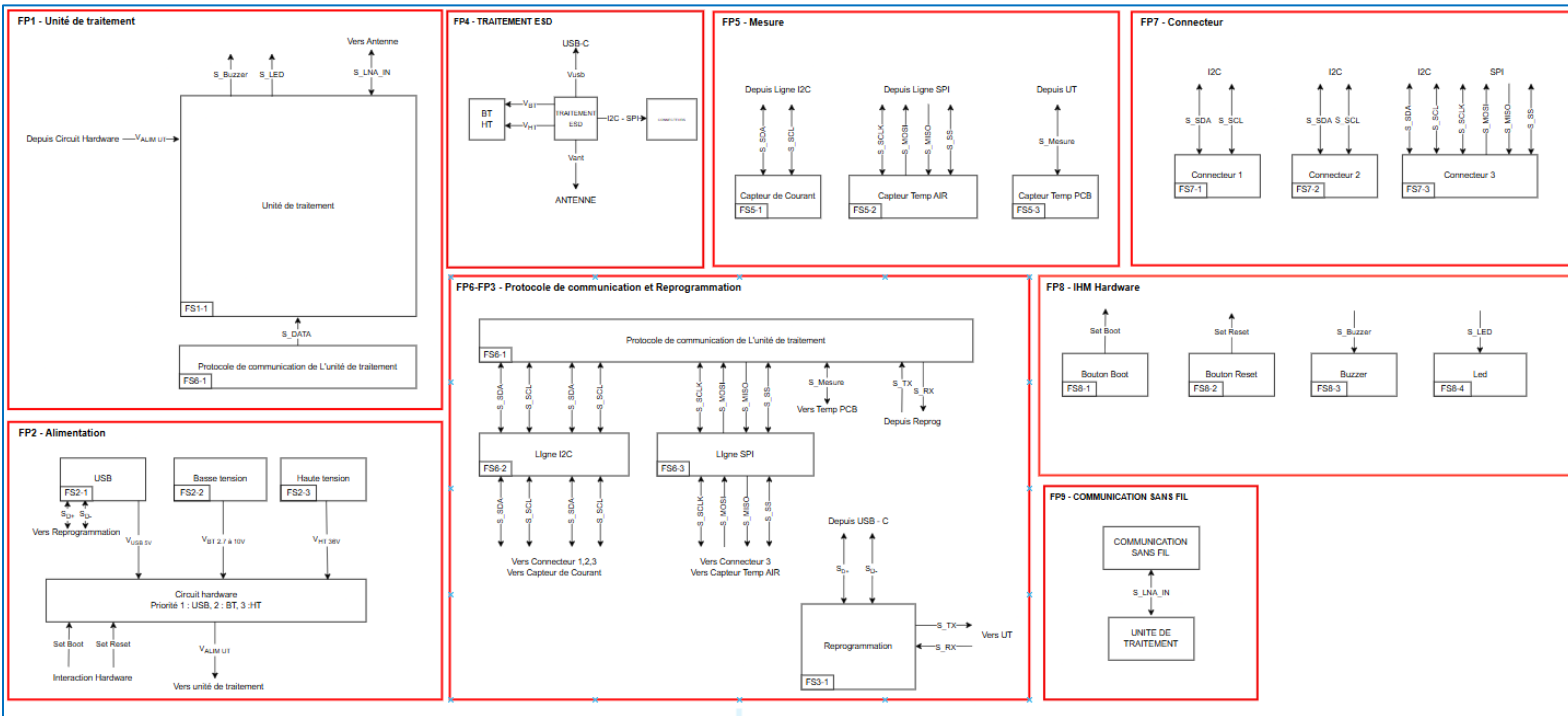
3. Interface Homme-Machine (IHM) : Utilisation d'une LED, d'un buzzer et de BLE pour signaler l'état et communiquer avec d'autres dispositifs.

4. Contrôle matériel : Boutons pour démarrage (boot) et réinitialisation (reset).

5. Compatibilité : Modules MikroBus et Grove ajoutent des extensions fonctionnelles.

Composants : ESP32-PICO-D4 (microcontrôleur), convertisseur DC-DC, diodes Schottky, capteur INA, TVS, LED, buzzer, et antenne BLE.

Analyse Fonctionnelle niveau 2



https://drive.google.com/file/d/1MQUc3rLA9nF23ld9bj9-Fkss09cnbmY_/view?usp=sharing

Explications Schéma

FP1 - Unité de traitement : Cette section représente l'unité centrale du système (ESP-32). Elle communique avec les autres parties du système, notamment pour la gestion des signaux de buzzers et de LEDs. Elle reçoit des informations des différents capteurs et modules via des protocoles de communication (FP6-FP3)

FP2 - Alimentation : Cette unité gère les trois sources d'alimentation de la carte avec les priorités suivantes :

1. USB-C : 5V
2. basse tension : 2,7V à 10V
3. haute tension : jusqu'à 36V

FP4 – Traitement ESD : Cette section est responsable de la protection contre l'électricité statique. Elle est présente dans tous les circuits sensibles à ce phénomène.

FP5 - Mesure : Elle est dédiée à l'intégration des capteurs. Cette unité effectue des mesures sur les courants et les températures de l'air et du PCB, envoyant les données à l'unité de traitement pour analyse.

FP6-FP3 - Protocole de communication et Reprogrammation : Cette partie assure les protocoles de communication entre les différents composants du système (via I2C, SPI, USB-C) et gère la reprogrammation du système, y compris la transmission de données entre l'unité de traitement et d'autres capteurs ou modules externes.

FP7 - Connecteur : Cette unité décrit la gestion des connecteurs physiques, permettant la communication avec différents modules via des lignes I2C et SPI. Chaque connecteur relie un module spécifique à l'unité de traitement.

FP8 - IHM (Interface Homme-Machine) Hardware : Elle représente les composants physiques permettant à l'utilisateur d'interagir avec le système, comme les boutons de réinitialisation (reset), démarrage (boot), ainsi que les signaux visuels et sonores via des LEDs et des buzzers.

FP9 - Communication sans fil : Cette unité gère la communication sans fil, via BLE (Bluetooth Low Energy), permettant la communication à distance avec d'autres dispositifs.

Problèmes prévus

- Réalisation du système de portes logiques
- Gestion de la RF
- Calculs des Résistances et Condensateurs

Schématiques

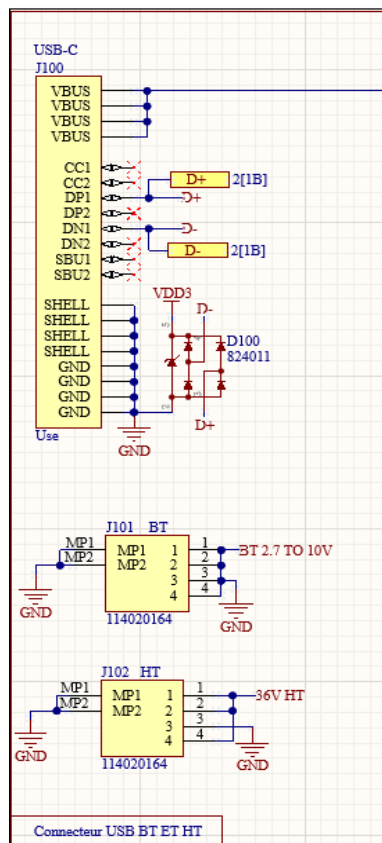


Figure 1 : Connecteurs USB BT et HT

Figure 1 : Le connecteur USB-C est protégé contre l'ESD avec les diodes TVS D100.

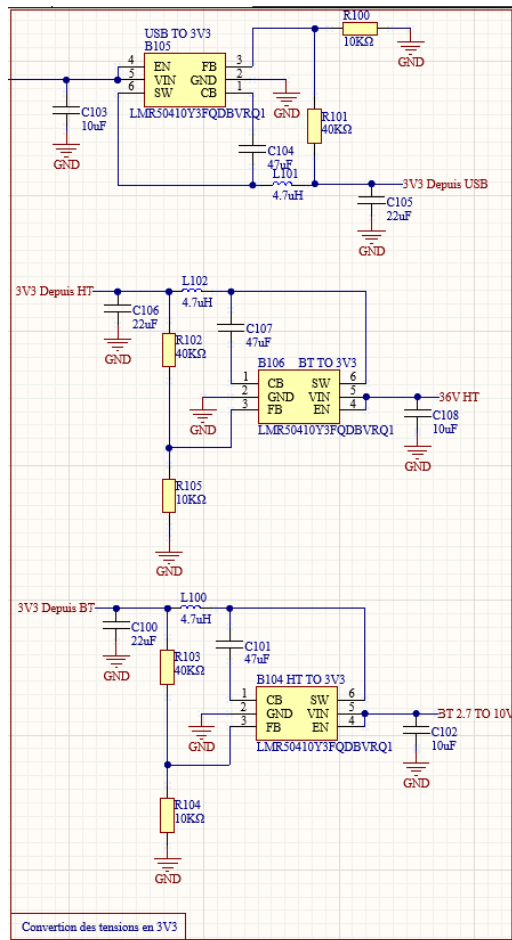


Figure 2 : Conversion des tensions en 3V3

Figure 2 : Les modules B104 – B105 – B106 régulent la tension des différentes entrées vers du 3V3, qui sera ensuite récupéré par le circuit de portes logiques (Figure 3).

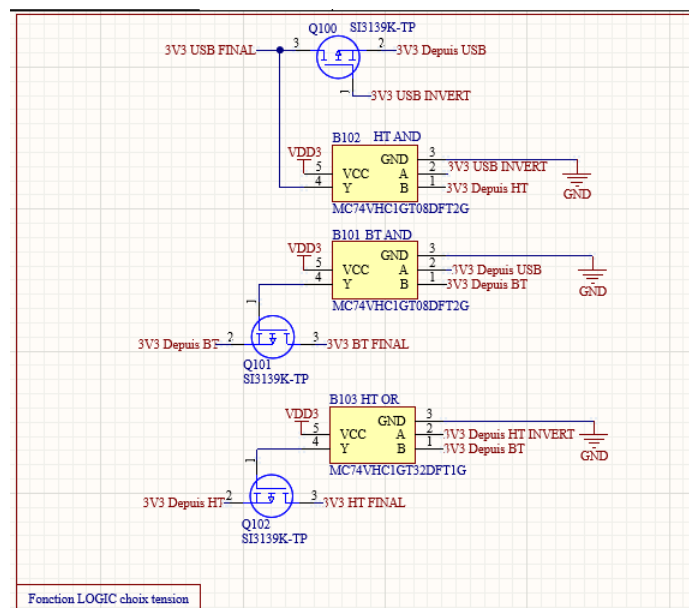


Figure 3 : Portes logiques choix Tension

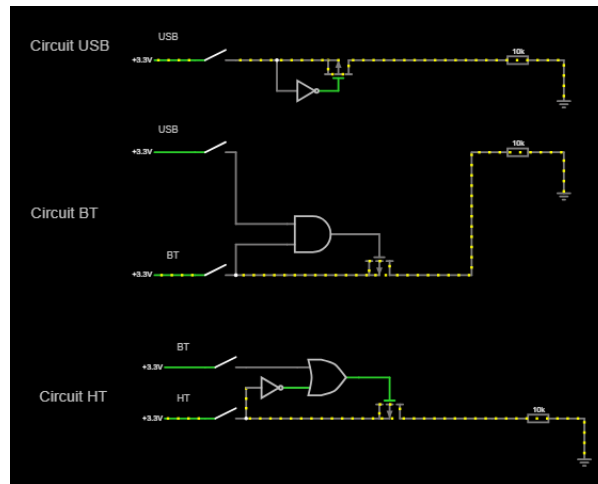


Figure 4 : Circuit logique Falstad

Figures 4 – 5 : Circuit logique comprenant des portes ET – OU, des Mosfet P et un inverseur de signal (USB-C et HT). <https://tinyurl.com/2c78ws6f>

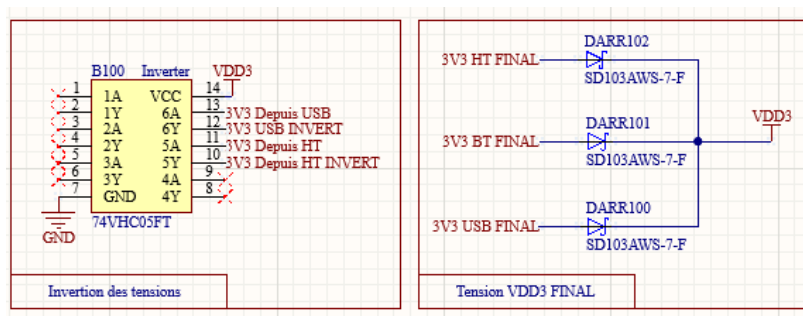


Figure 5 : Inversion tension et VDD3 Final

Figure 5 : L'inverseur pour les portes logiques, ainsi que la sortie des tensions donnant VDD3.

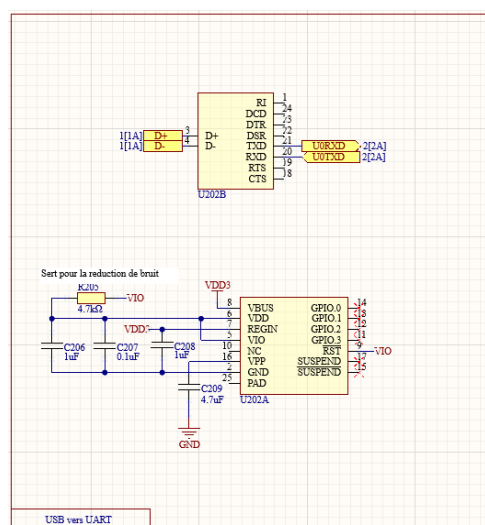


Figure 6 : USB vers UART

Figure 6 : Cette partie transforme le signal USB vers l'UART avec l'USB Bridge. Les capas sont utilisées pour réduire le bruit émanant de l'USB.

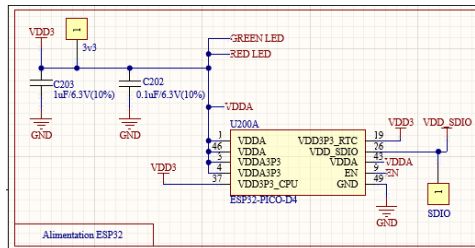


Figure 7 : Alimentation ESP32

Figure 7 : alimentation de l'ESP32 avec les filtrages en amont, et l'alimentation des LEDs verte et rouge en fonction de l'état du système.

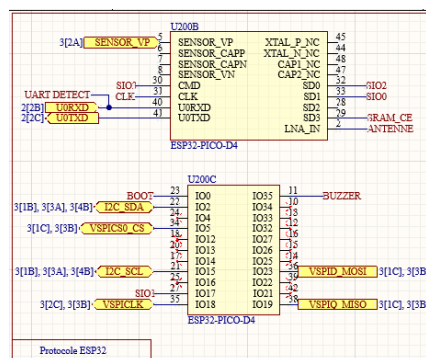


Figure 8 : Protocole ESP32

Figure 8 : Les pins utilisées pour les différents capteurs et protocoles de communications.

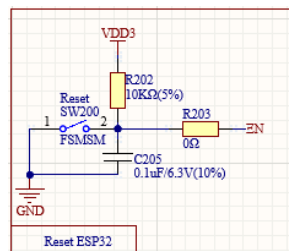


Figure 9 : Reset ESP32

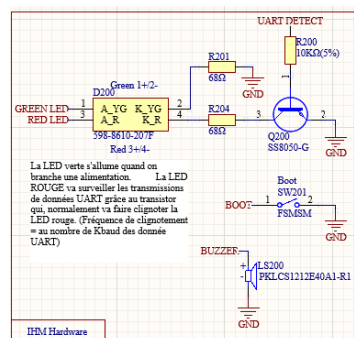


Figure 10 : IHM Hardware

Figure 10 : Circuit d'alimentation des LEDs rouge et verte ; du Buzzer et du bouton Boot.

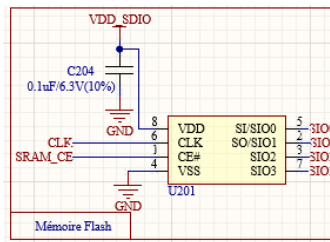


Figure 11 : Mémoire flash

Figure 11 : Gestion de la mémoire Flash avec le SRAM de 8mb obligatoire pour l'ESP32 que l'on a produit.

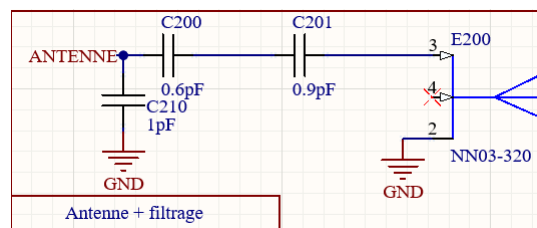


Figure 12 : Antenne + Filtrage

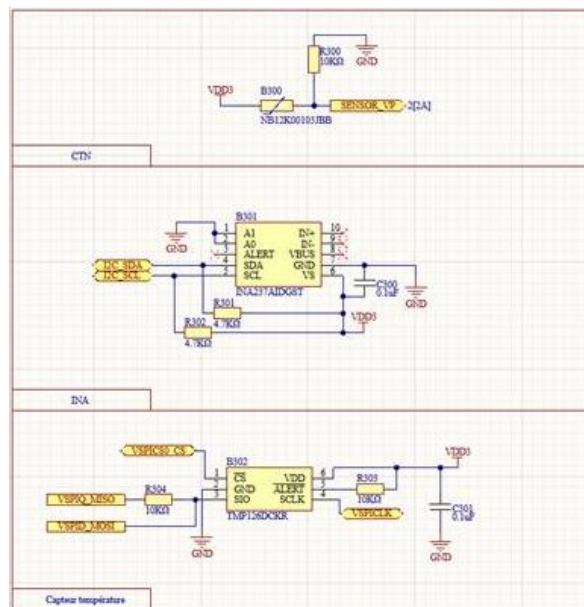


Figure 13 : Capteurs

Figure 13 : Les capteurs de température d'air, du PCB et de l'ESP32 avec des fonctionnements différents (Courant pour l'INA, résistance pour la CTN et avec une diode pour celui de l'air ambiant).

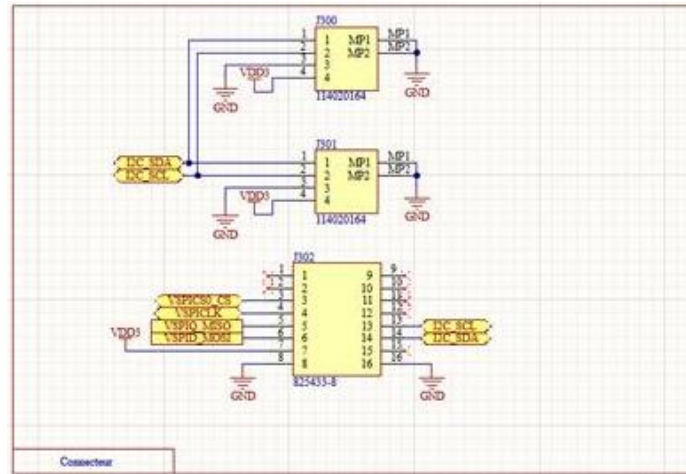


Figure 14 : Connecteurs

Routage

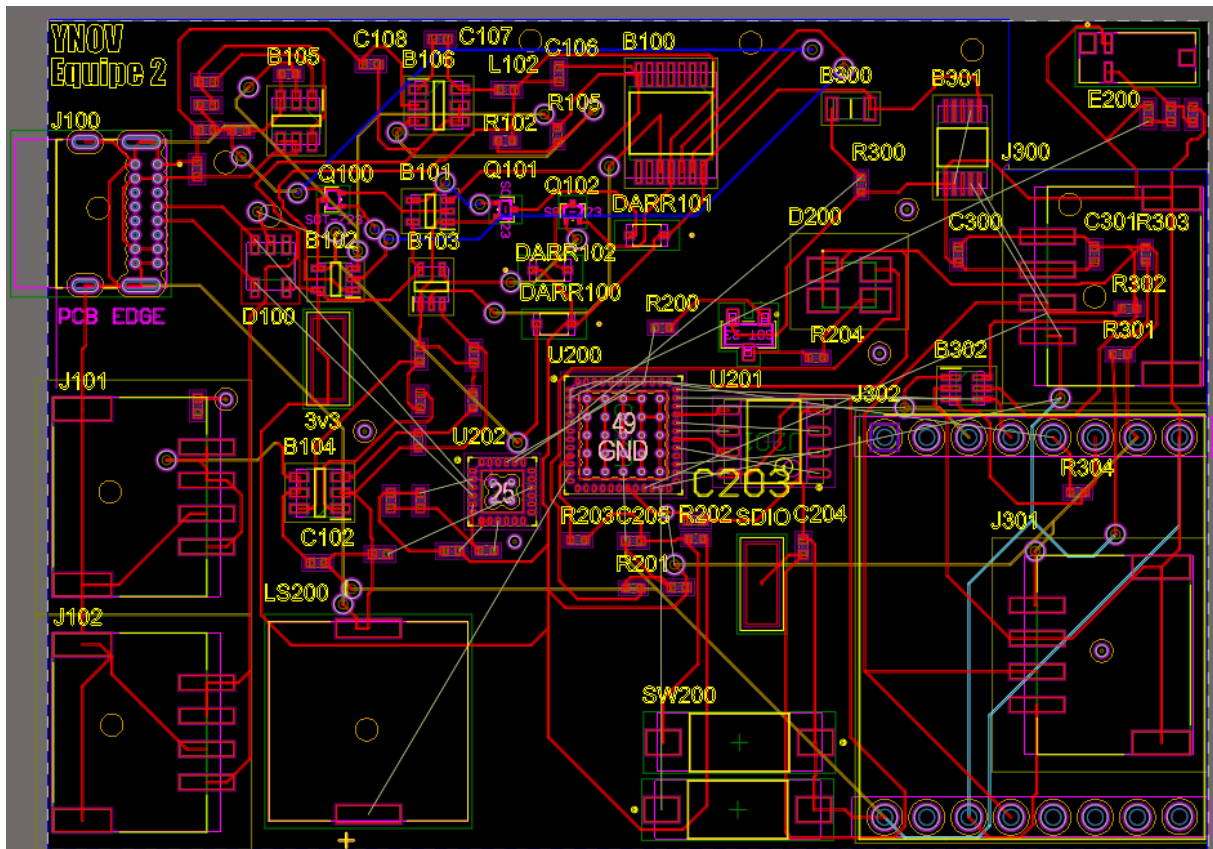


Figure 15 : Routage PCB

Figure 15 : représentation du PCB à 70 % finit. Les connecteurs sont disposés de part et d'autre pour plus de praticité.

Il y a 4 plans :

- 1 plan de masse (GND)
- 1 plan VCC (3v3)
- 1 plan I2C
- 1 plan SPI

En haut à droite, la carte est dépourvue de plan pour ne pas avoir de bruit généré par l'antenne. Sur le routage des signaux UART, I2C et SPI il y a une règle très importante : il faut que les pistes de DATA+ et DATA- soient de la même longueur pour éviter un décalage et de ne pas pouvoir communiquer avec l'ESP32. Cette règle de routage est faite pour les hautes fréquences, elle ne nous concerne pas.

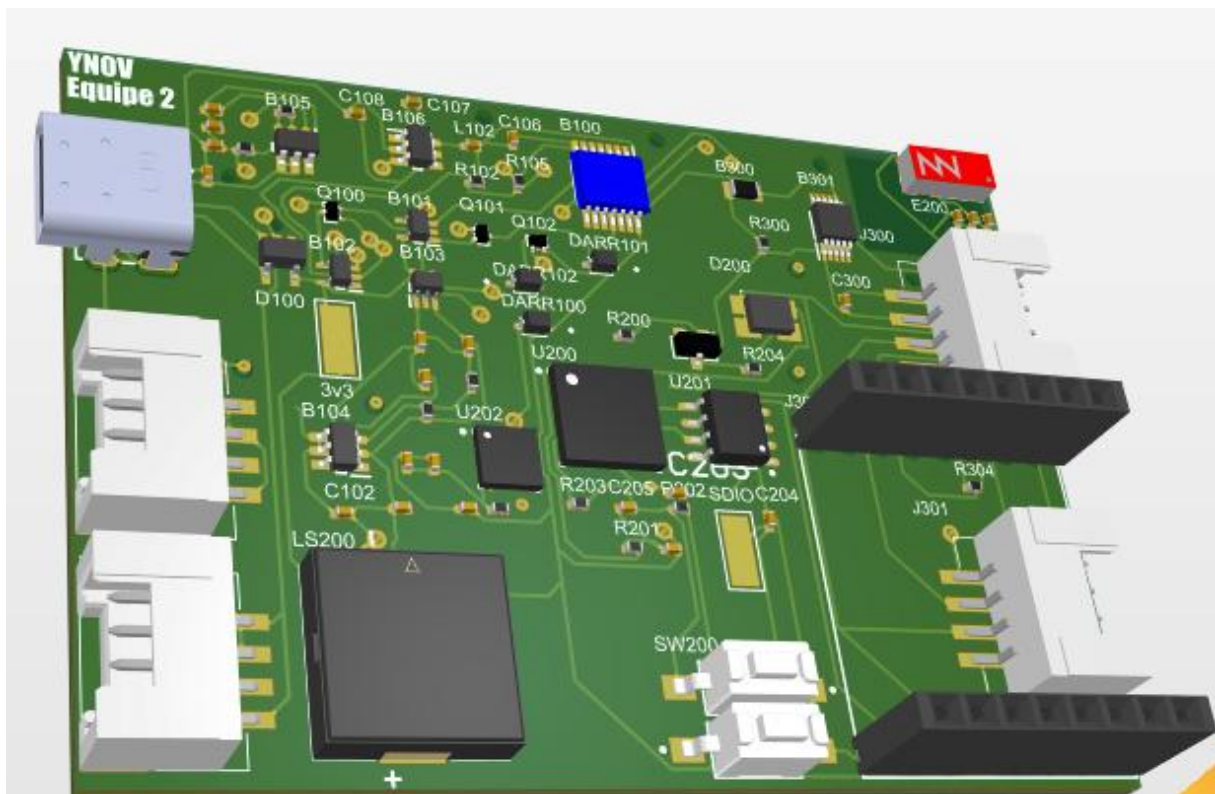


Figure 6 : Représentation carte final

Voici la représentation de la carte une fois construite grâce au fichier gerber x2.

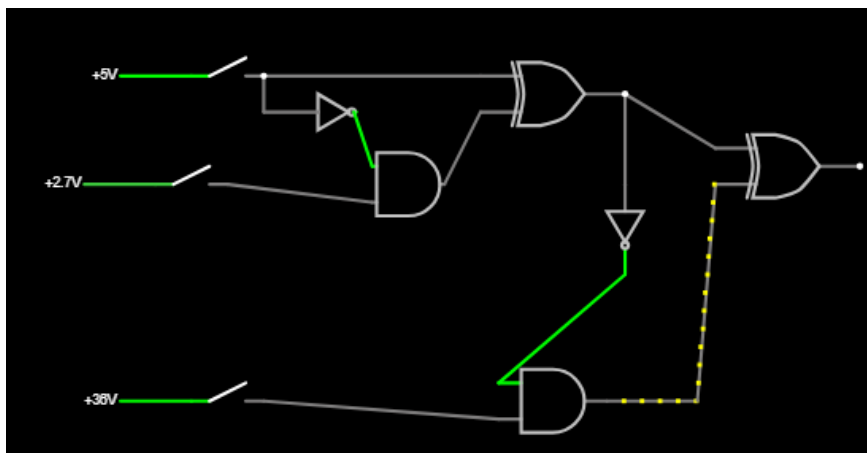
Problème rencontré 1

Présentation du problème

La réalisation du circuit logique a constitué un obstacle important lors de la création des schématiques. L'utilisation de Falstad a été obligatoire, mais ce dernier était limité dans sa gestion des priorités. Une division du système a donc été effectuée pour 3 raisons :

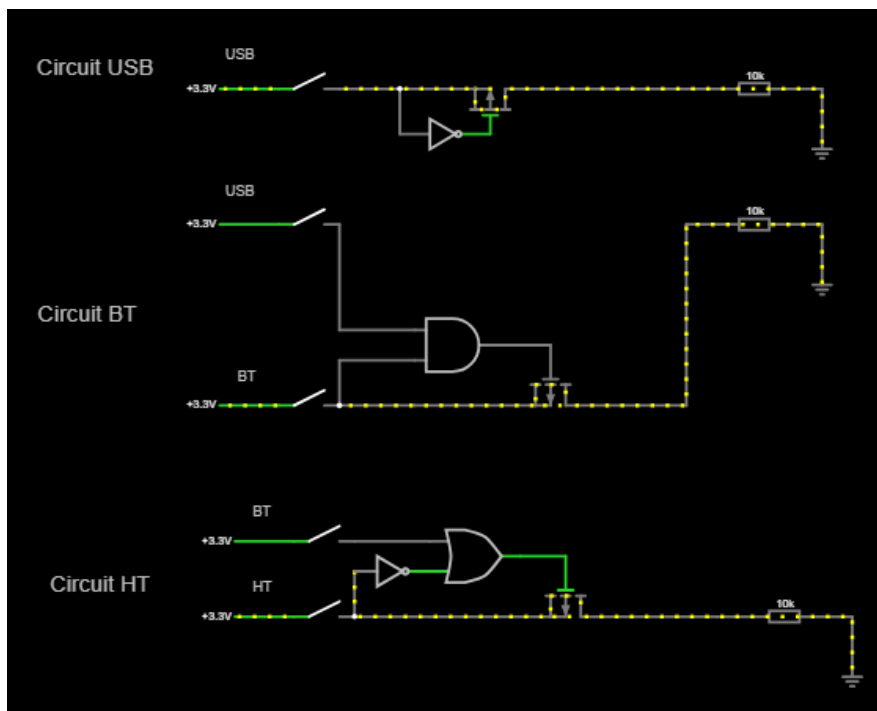
- Augmenter la lisibilité
- Simplifier le schéma
- Vérifier sur Falstad que le circuit fonctionne

Avant séparation :



Solution

Après séparation :



Le schéma est clair, et fonctionnel. Nous avons rajouté des Mosfet P afin de remplacer les portes « ou exclusifs » précédemment utilisées.

Problème rencontré 2

Présentation du problème

Le routage PCB sur 4 couches comment cela fonctionne par rapport aux signaux (I2C et SPI).

Solution

Faire plusieurs plans au même niveau de couche et passer par des VIA.
Cela reste encore Ambiguë par rapport aux exemples vu sur internet.

Conclusion Projet

La partie analyse fonctionnel était assez simple et relativement rapide à faire.
Sur la schématique les recherches de DATASHEET étaient fastidieuses au début en raison de nombreuses informations à trier et comprendre.

Le Niveau de l'équipe 2 était assez hétérogène, doublé d'une disparité de langues. Ce n'était donc pas simple d'expliquer et avancer en même temps (comme le ferai des chefs de projets).