

RAPPORT PROJET CARTE DE DÉVELOPPEMENT ESP32

Réalisées par :

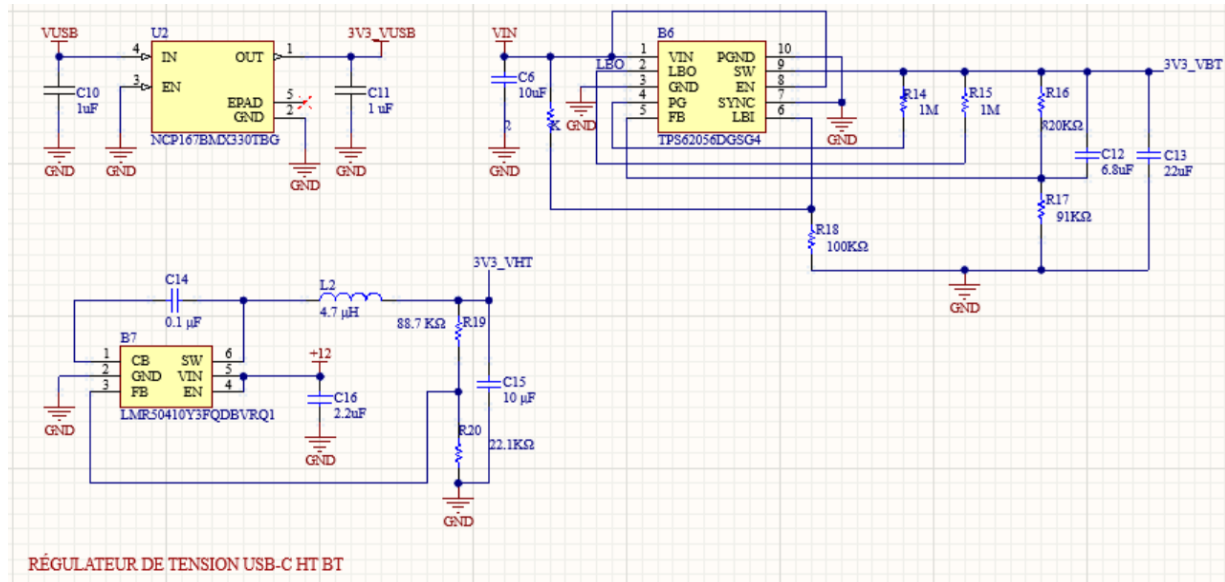
Alexis Toribio

Nassima Elotmani

Albouy Thomas

Encadré par : Jordan Clement

Circuit régulateur de tension



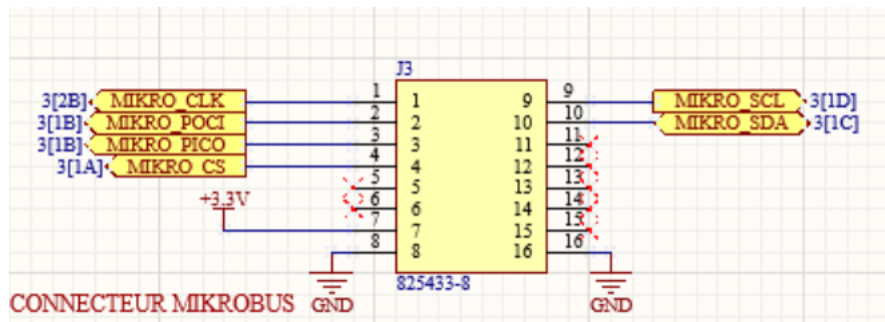
Ce schéma représente un circuit régulateur de tension basé sur plusieurs modules de conversion et régulation. Il permet de générer différentes tensions stables à partir d'une source d'alimentation USB-C, utile pour alimenter divers sous-systèmes électroniques.

Le schéma propose trois circuits régulateurs de tension :

- U2 (NCP167BM) : Régulateur LDO pour convertir l'alimentation USB en 3.3V_VUSB avec faible bruit.
- B6 (TPS62056) : Convertisseur DC-DC abaisseur pour générer 3.3V_VBT de manière efficace.
- B7 (LMR50410) : Deuxième convertisseur DC-DC pour fournir 3.3V_VHT à partir de 12V avec un rendement élevé.

Chaque partie assure une régulation de tension fiable tout en minimisant les pertes d'énergie et les bruits électriques. Ce schéma est adapté pour des applications nécessitant plusieurs rails de tension stables, comme des systèmes embarqués ou des circuits à microcontrôleur.

Connecteur Microbus



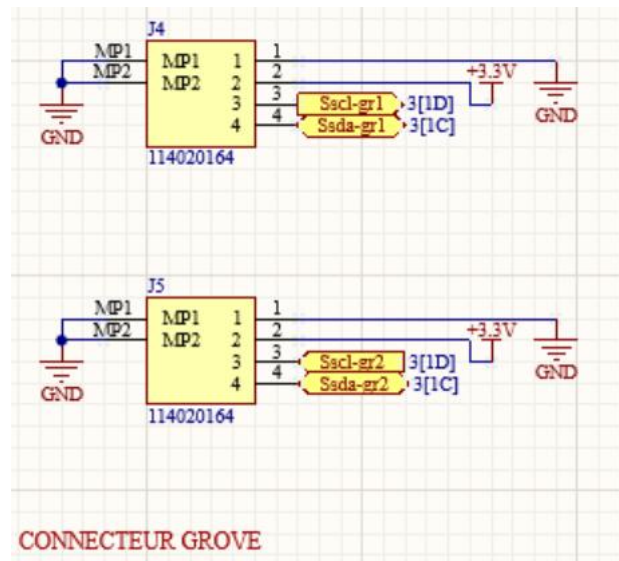
Cette partie du schéma représente **un connecteur MikroBUS**, utilisé pour interfacier des modules MikroE avec un circuit principal, comme une carte mère ou un microcontrôleur. Le MikroBUS est un standard populaire pour connecter facilement des capteurs, actionneurs ou autres modules via des interfaces et **I2C**.

Le connecteur MikroBUS (J3) est conçu pour faciliter l'interconnexion des modules MikroE avec le circuit principal via des interfaces standards comme SPI et I2C. Voici les rôles principaux :

- Communication SPI (Broches 1 à 4) pour des échanges rapides avec des périphériques nécessitant une communication synchrone.
- Alimentation 3.3V (Broche 6) pour alimenter les modules connectés.
- Communication I2C (Broches 10 et 11) pour des périphériques à bas débit comme des capteurs.

Ce connecteur standardisé simplifie le développement et l'intégration de périphériques externes sur des systèmes embarqués ou des cartes électroniques.

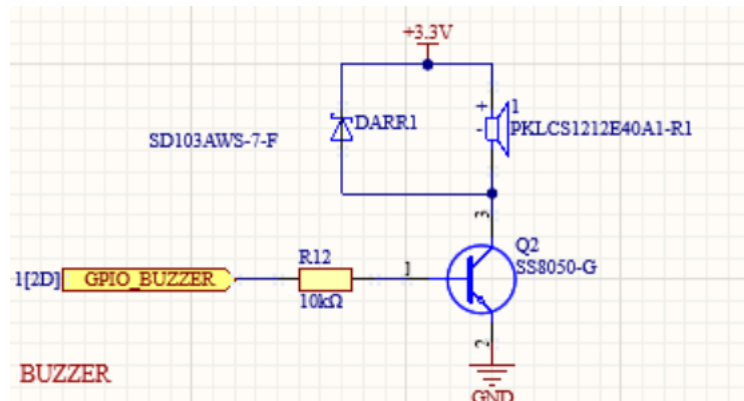
connecteurs Grove



Cette partie du schéma représente deux connecteurs Grove (J4 et J5) permettant d'établir des connexions standardisées pour des modules compatibles avec le système Grove. Grove est une plateforme modulaire utilisée dans le prototypage rapide pour connecter facilement des capteurs, des actionneurs ou autres périphériques électroniques à un circuit principal, généralement via une interface I2C.

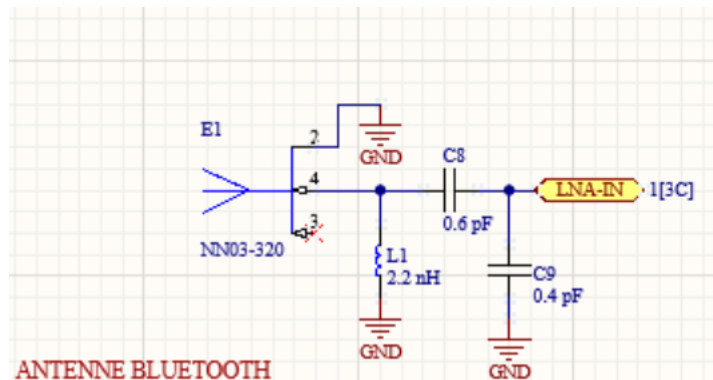
Cette partie du schéma permet l'interfaçage de deux modules Grove I2C via les connecteurs J4 et J5 :

- Communication I2C :
 1. SCL pour l'horloge série (synchronisation).
 2. SDA pour les données série.
- Alimentation en 3.3V pour les modules connectés.
- Connexion simplifiée via le standard Grove pour des capteurs et périphériques compatibles.



Ce schéma électrique représente un circuit de commande pour un buzzer actif contrôlé par un signal GPIO , généralement provenant d'un microcontrôleur ou d'un microprocesseur.

Ce circuit permet de contrôler un buzzer actif via un signal GPIO en utilisant un transistor comme interrupteur. La résistance R12 protège la sortie GPIO, tandis que la diode DARR1 protège le transistor contre les surtensions dues à la nature inductive du buzzer. Ce type de montage est courant dans les systèmes embarqués pour signaler un événement sonore, comme une alarme ou une notification.



Ce schéma électrique représente l'interface entre une antenne Bluetooth et un circuit d'entrée faible bruit , qui est typiquement utilisé dans les systèmes de communication sans fil. Voici une explication détaillée pour votre rapport technique.

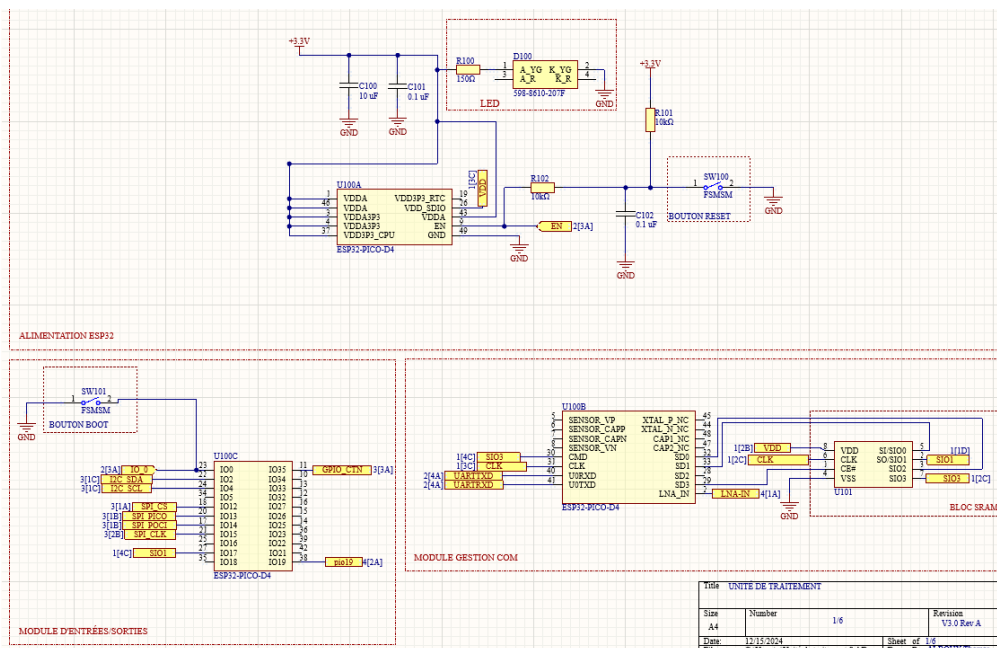
Ce circuit est une interface d'adaptation d'impédance et de filtrage entre une antenne Bluetooth et un amplificateur faible bruit (LNA). L'objectif principal est de :

- Maximiser la transmission du signal RF reçu par l'antenne.
- Filtrer les bruits indésirables.
- Fournir un signal propre et amplifiable au LNA pour améliorer la performance de réception.

Ce type de circuit est essentiel dans les dispositifs Bluetooth pour garantir une réception efficace des signaux RF

Lors de l'intégration de l'antenne NN03-320 dans le circuit, une difficulté rencontrée a été l'identification de la fonction des trois broches (pins) de l'antenne. Les informations précises concernant ces broches n'étaient pas facilement accessibles dans la documentation technique. Après une analyse approfondie et des recherches complémentaires, il a été déterminé que :

- La broche 4 est la sortie RF de l'antenne, par laquelle le signal est transmis au circuit suivant.
- La broche 3 et la broche 2 sont connectées à la masse (GND) pour assurer une référence électrique stable et optimiser les performances de l'antenne.



Ce schéma électronique représente une carte de développement basé autour d'un microcontrôleur ESP32-PICO-D4. Il est divisé en plusieurs sections principales :

- **Alimentation** : Cette section gère l'alimentation du circuit à partir d'une source externe (probablement 5V ou 3.3V).

- **Module de gestion de la communication** : Cette partie comprend l'ESP32-PICO-D4 lui-même, qui est le cœur du système. Il gère les communications (Wi-Fi, Bluetooth), les entrées/sorties et les calculs.
- **Module d'entrées/sorties** : Cette section contient les différents composants qui permettent à la carte d'interagir avec le monde extérieur : boutons, LED, capteurs, etc.
- **Unité de traitement** : Cette section est moins détaillée, mais elle pourrait faire référence à une partie spécifique du microcontrôleur ou à un bloc fonctionnel plus large.

Analyse détaillée (éléments clés)

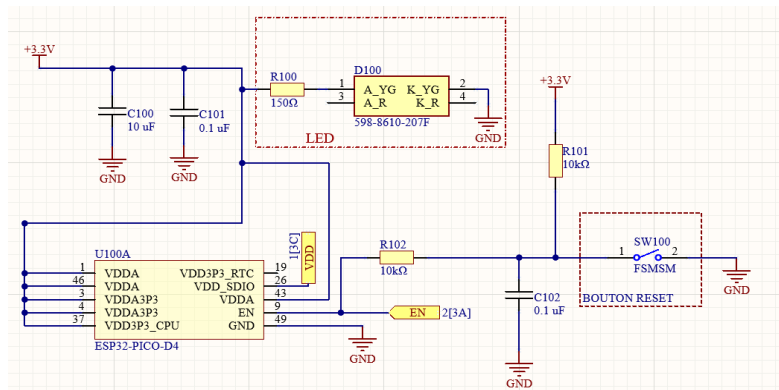
- **ESP32-PICO-D4** : C'est un microcontrôleur à faible consommation d'énergie, très populaire dans le monde de l'IoT. Il offre des fonctionnalités Wi-Fi et Bluetooth, ainsi qu'un grand nombre de broches d'E/S.
- **Composants passifs** : Résistances, condensateurs... Ils servent à conditionner les signaux, filtrer le bruit, etc.
- **Boutons** : Ils permettent à l'utilisateur d'interagir avec le système.
- **LED** : Elles servent d'indicateurs visuels.
- **Autres composants (à identifier)** : Selon les références, il pourrait s'agir de capteurs, d'actionneurs, ou d'autres composants spécifiques à l'application.

Fonctionnement probable

Sans plus d'informations sur le contexte de ce circuit, il est difficile de décrire précisément son fonctionnement. Cependant, on peut émettre quelques hypothèses

- **Initialisation** : Au démarrage, le microcontrôleur lit les valeurs des différents composants (boutons, capteurs) et configure les sorties (LED, actionneurs).
- **Boucle principale** : Le microcontrôleur exécute en boucle un programme qui vérifie les entrées, effectue des calculs et met à jour les sorties.

Communications : L'ESP32-PICO-D4 peut se connecter à un réseau Wi-Fi ou Bluetooth pour communiquer avec d'autres appareils ou un serveur



Fonctionnement général

Analyse détaillée du circuit ALIMENTATION ESP32

Alimentation :

- **Entrée de tension :** Le circuit reçoit une tension continue de 3.3V. Cette tension est généralement fournie par une source d'alimentation externe, comme un adaptateur secteur ou une batterie.
- **Distribution :** Cette tension est distribuée à différents composants du circuit, y compris le microcontrôleur ESP32-PICO-D4.

Filtrage :

- **Rôle des condensateurs :** Les condensateurs C100 et C101 agissent comme des filtres passe-bas. Ils permettent de lisser la tension en atténuant les fluctuations de tension qui peuvent être causées par l'alimentation ou par d'autres composants du circuit.
- **C100 :** Ce condensateur, généralement de plus grande capacité, filtre les basses fréquences. Il stabilise la tension continue en absorbant les variations de courant.
- **C101 :** Ce condensateur, de plus petite capacité, filtre les hautes fréquences. Il réduit le bruit électrique qui pourrait perturber le fonctionnement du circuit.

Protection :

- **Diode Zener (D100) :** Cette diode est polarisée en inverse. Si la tension d'entrée dépasse une certaine valeur (la tension Zener de la diode), la diode se met en conduction, shuntant ainsi l'excès de tension vers la masse. Cela protège les composants sensibles du circuit contre les surtensions.
- **Résistance R100 :** Cette résistance limite le courant qui traverse la diode Zener lorsque celle-ci est en conduction, évitant ainsi une surchauffe de la diode.

Contrôle :

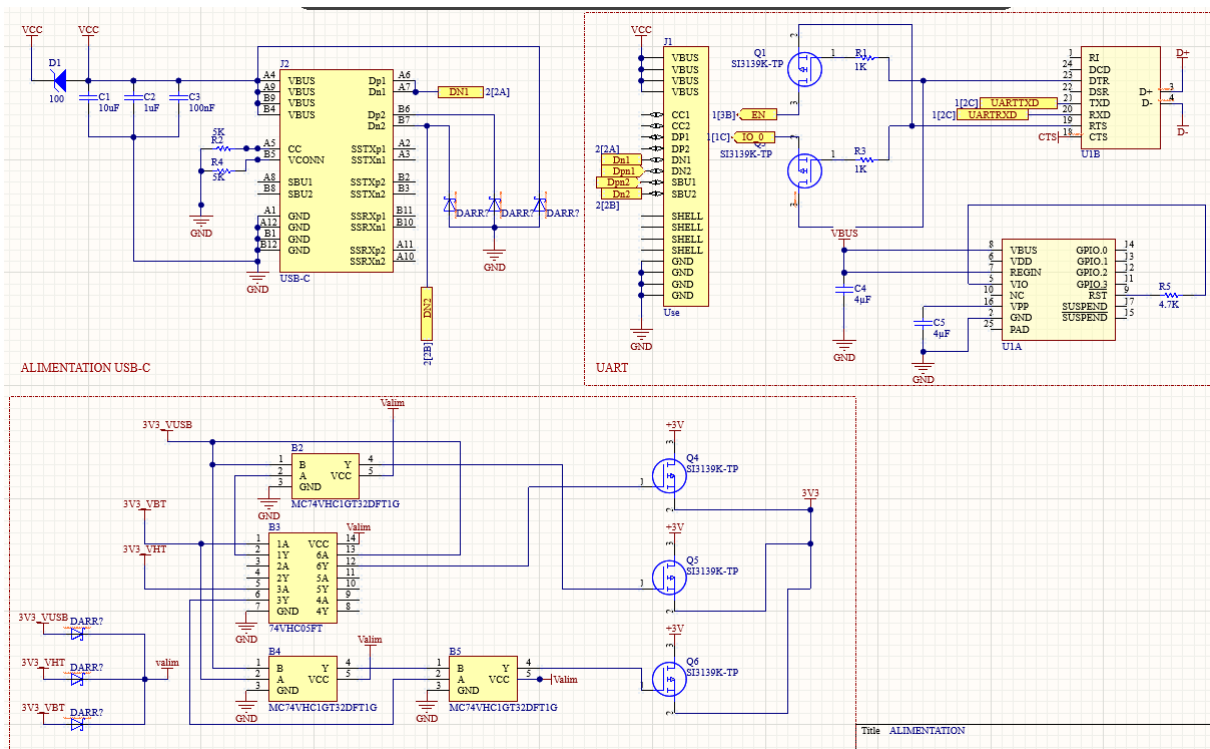
- **Transistor SW100** : Ce transistor, généralement un MOSFET, agit comme un interrupteur contrôlé. Il est piloté par le microcontrôleur.
- **Fonctionnement** : Lorsque le microcontrôleur envoie un signal approprié à la grille du MOSFET, celui-ci conduit le courant et permet ainsi à l'alimentation d'atteindre une partie spécifique du circuit. Inversement, lorsqu'il n'y a pas de signal, le MOSFET est bloqué et l'alimentation est coupée.
- **Applications** : Le transistor SW100 peut être utilisé pour :
 - Mettre sous tension ou hors tension des composants.
 - Réguler la puissance fournie à certains composants.
 - Implémenter des fonctions de protection (par exemple, couper l'alimentation en cas de surcharge).

Indication :

- **LED** : La LED sert d'indicateur visuel de l'état du circuit. Lorsqu'elle est allumée, cela signifie généralement que le circuit est alimenté et fonctionne correctement.
- **Fonctionnement** : La LED est connectée en série avec une résistance (non représentée sur le schéma) pour limiter le courant qui la traverse. Le courant passe à travers la LED lorsqu'elle est connectée à une tension suffisante.

En résumé

Ce bloc de circuit assure l'alimentation électrique du microcontrôleur ESP32-PICO-D4 en fournissant une tension stable et protégée. Il permet également de contrôler l'alimentation de différents composants du circuit grâce au transistor SW100. La LED sert d'indicateur visuel de l'état du système.



Le schéma présenté semble complexe au premier abord, mais il s'agit essentiellement d'un circuit d'alimentation conçu pour fournir une tension stable et sécurisée à un dispositif électronique, peut-être un ESP32-PICO-D4.

- **Plusieurs sources d'alimentation:** Le schéma indique la présence de plusieurs entrées de tension, ce qui correspond à l'objectif de votre projet.
- **Régulateurs de tension:** Les différents régulateurs permettent de transformer les tensions d'entrée en tensions plus faibles et plus stables.
- **Protection:** Les diodes et les résistances sont probablement utilisées pour protéger les composants contre les surtensions et les courts-circuits.
- **Interface utilisateur:** Bien que non explicite sur le schéma fourni, la présence d'un bouton de réinitialisation suggère une interface utilisateur basique.

Adaptation du schéma au projet

Pour adapter ce schéma à votre projet, plusieurs modifications et ajouts sont nécessaires :

1. Priorisation des sources d'alimentation:

- **Détection automatique:** Implémenter un circuit de détection automatique pour déterminer la source d'alimentation active en fonction de la priorité (USB, BT, HT).

- **Commutation:** Utiliser des MOSFET ou des relais pour commuter entre les différentes sources en fonction de la détection.
- 2. Mesure de courant et de température:**
- **INA:** Intégrer un circuit INA (Integrated Current Sensor) pour mesurer le courant consommé par le système.
 - **Capteur de température:** Ajouter un capteur de température (thermistance, capteur numérique) pour mesurer la température de fonctionnement.
- 3. Interface utilisateur:**
- **Buzzer:** Connecter le buzzer au microcontrôleur pour générer des signaux sonores en fonction des événements (démarrage, erreur, etc.).
 - **LED:** Utiliser la LED pour indiquer l'état du système (alimentation, erreur, mode de fonctionnement).
 - **Boutons:** Configurer les boutons de démarrage et de réinitialisation pour envoyer les signaux appropriés au microcontrôleur.
- 4. Communication BLE:**
- **Module BLE:** Ajouter un module Bluetooth Low Energy (BLE) à l'ESP32-PICO-D4 pour permettre la communication sans fil.
 - **Antenne:** Connecter une antenne adaptée au module BLE.
- 5. Connecteurs MikroBus et Grove:**
- **Brochage:** Assurer une connexion correcte entre les broches de l'ESP32 et les connecteurs MikroBus et Grove.
 - **Protocoles:** Configurer les protocoles de communication (I2C, SPI, UART) pour les modules connectés.

Schéma simplifié proposé

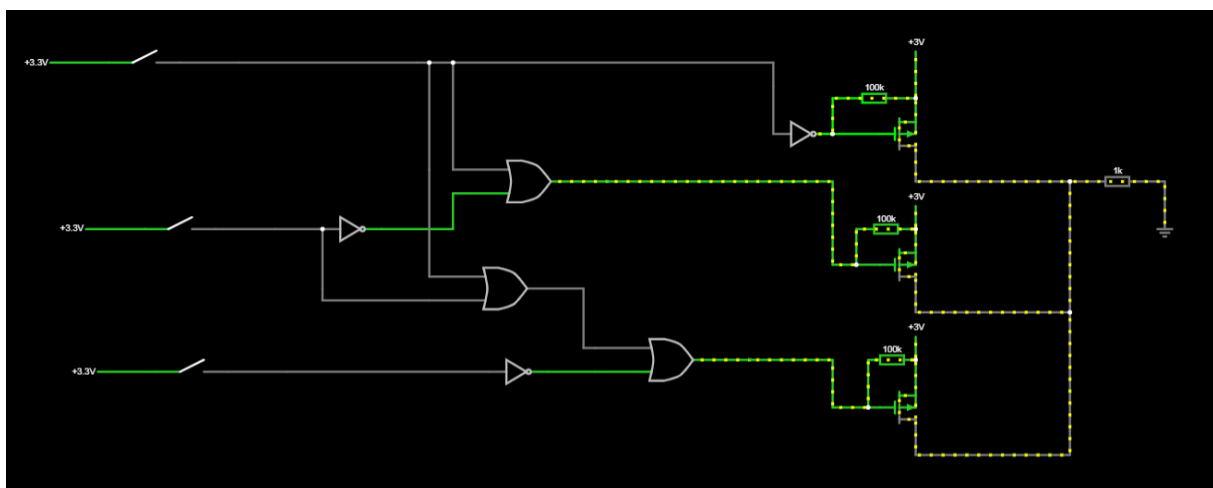
[Image d'un schéma simplifié avec les éléments suivants :

- **Entrée USB-C:** Connectée à un régulateur buck pour obtenir 5V.
- **Entrée BT:** Connectée à un régulateur buck pour obtenir 3.3V (adaptable en fonction de la tension d'entrée).
- **Entrée HT:** Connectée à un régulateur buck pour obtenir 3.3V.
- **Diodes de protection:** Pour chaque entrée, afin d'éviter les inversions de polarité.

- **MOSFET:** Pour commuter entre les différentes sources d'alimentation.
- **Circuit de détection:** Pour déterminer la source active.
- **Régulateur linéaire 3.3V:** Pour alimenter l'ESP32-PICO-D4.
- **INA:** Pour mesurer le courant.
- **Capteur de température:** Connecté à une entrée analogique de l'ESP32.
- **Buzzer:** Connecté à une sortie PWM de l'ESP32.
- **LED:** Connectée à une sortie numérique de l'ESP32.
- **Module BLE:** Connecté aux broches SPI de l'ESP32.
- **Connecteurs MikroBus et Grove:** Connectés aux broches GPIO de l'ESP32.]

Test Falstad : <https://tinyurl.com/29gkbljc>

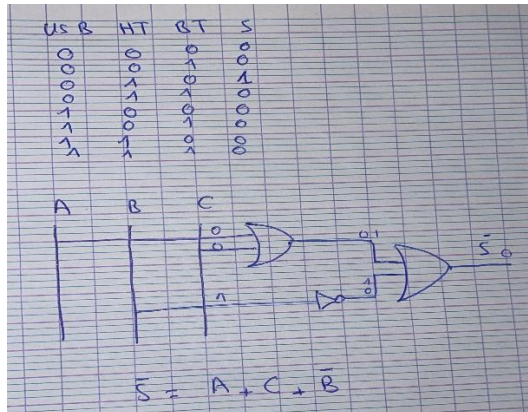
<https://tinyurl.com/294pscq4>



La logique :

- **USB :** 1 si l'alimentation USB est présente, 0 sinon.
- **BT :** 1 si l'alimentation BT est présente, 0 sinon.
- **HT :** 1 si l'alimentation HT est présente, 0 sinon.

- **Demonstration :**



- **A, B, C :** Ces variables sont probablement des sorties intermédiaires de portes logiques utilisées pour construire la fonction de sortie S.
- **Sortie S:**
 - **S** représente l'état de sortie du circuit. Une valeur '1' pourrait indiquer qu'une alimentation spécifique est sélectionnée, tandis qu'une valeur '0' pourrait indiquer qu'aucune alimentation n'est sélectionnée ou qu'il y a une erreur.

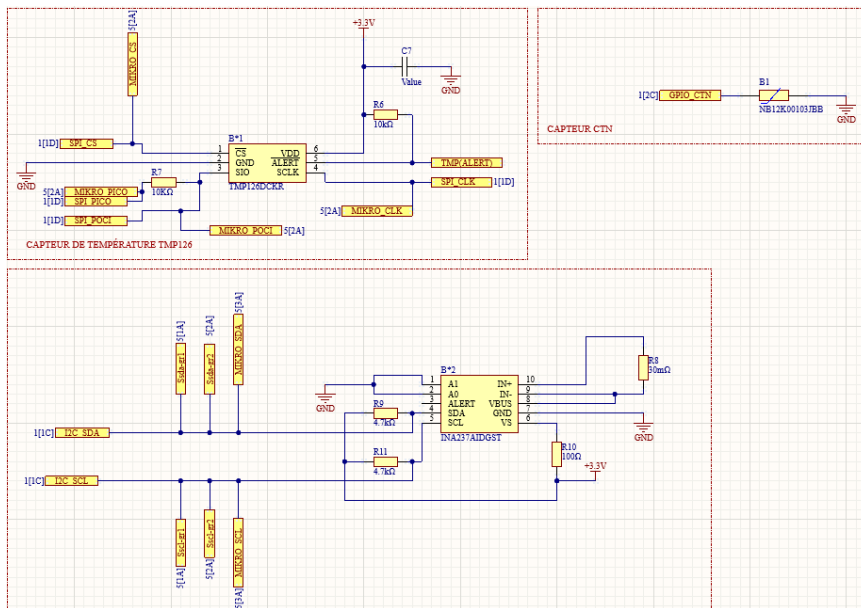
Interprétation de la Logique

L'équation logique $S = A + C + B'$ suggère ce qui suit :

- **A, B, C:** Ces variables sont liées à des conditions spécifiques sur les entrées USB, BT et HT. Par exemple, A pourrait représenter la condition "USB est présente", B pourrait représenter "BT est présente et USB est absente", et C pourrait représenter "HT est présente et USB et BT sont absentes".
- **Priorité:** L'opérateur "+" (OU) indique que si l'une des conditions A, C ou B' est vraie, alors S sera vrai. Cela signifie que l'alimentation correspondante sera sélectionnée. Le terme B' (NON B) suggère que l'alimentation BT est sélectionnée seulement si les alimentations USB et HT ne sont pas disponibles.

Table de Vérité

La table de vérité fournit une représentation exhaustive de toutes les combinaisons possibles des entrées USB, BT et HT, ainsi que la valeur correspondante de la sortie S. En analysant cette table, on peut déterminer les conditions exactes pour lesquelles chaque alimentation est sélectionnée.



Le schéma présenté décrit un circuit électronique conçu pour mesurer la température à l'aide d'un capteur TMP126 et de mesurer le courant à l'aide d'un capteur INA237. Ces deux capteurs sont interfacés avec un microcontrôleur (non spécifié sur le schéma) via une interface I2C.

Fonctionnement Détaillé

- **Capteur de Température TMP126 :**

- Le TMP126 est un capteur de température numérique précis qui communique via une interface I2C.
- Il mesure la température et la convertit en un signal numérique qui est ensuite transmis au microcontrôleur.
- Les broches VDD et GND du TMP126 sont connectées à l'alimentation du circuit.
- Les broches SCL et SDA sont connectées aux bus I2C du microcontrôleur.

- **Capteur de Courant INA237 :**

- L'INA237 est un capteur de courant et de puissance haute précision qui mesure le courant continu.
- Il fournit des informations sur le courant, la tension et la puissance consommées par une charge.
- Les broches VBUS, GND, SDA et SCL sont connectées à l'alimentation et au bus I2C du microcontrôleur.

- La broche ALERT peut être utilisée pour générer une interruption lorsque le courant dépasse une certaine valeur seuil.

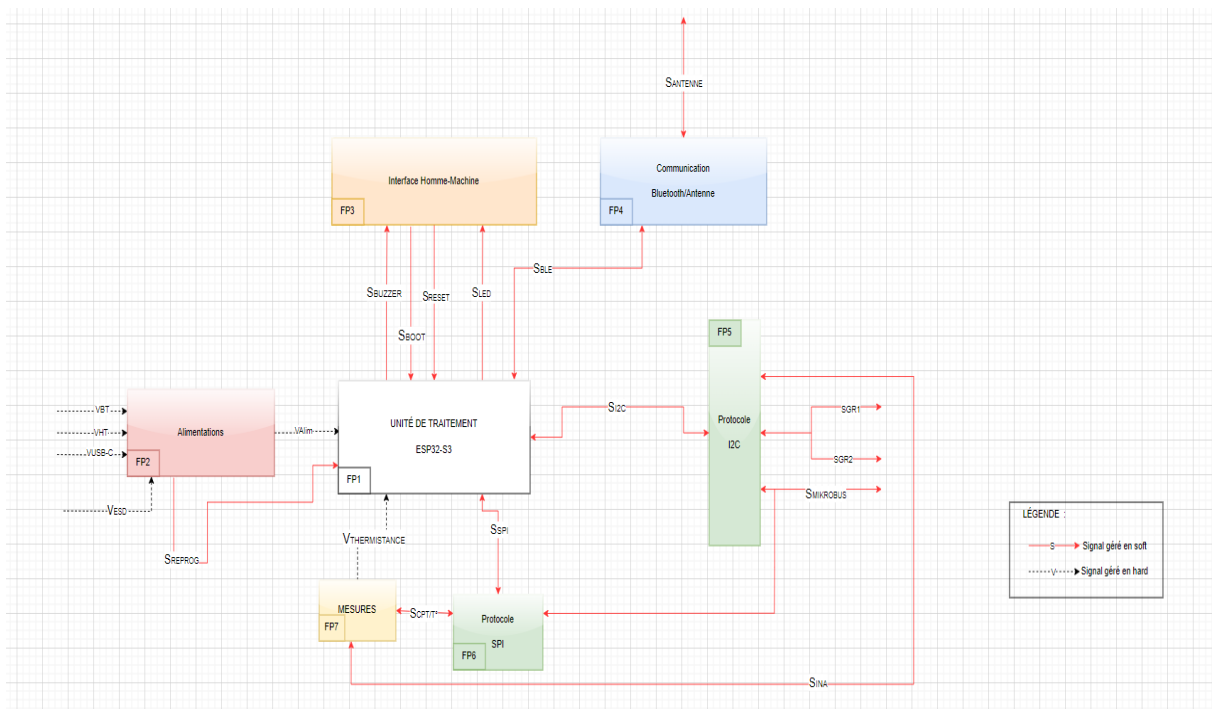
- **Interface I2C :**

- Les deux capteurs communiquent avec le microcontrôleur via l'interface I2C.
- L'interface I2C est un bus série bidirectionnel qui permet de connecter plusieurs périphériques sur le même bus.
- Le microcontrôleur utilise des registres spécifiques pour lire les données des capteurs et configurer leur fonctionnement.

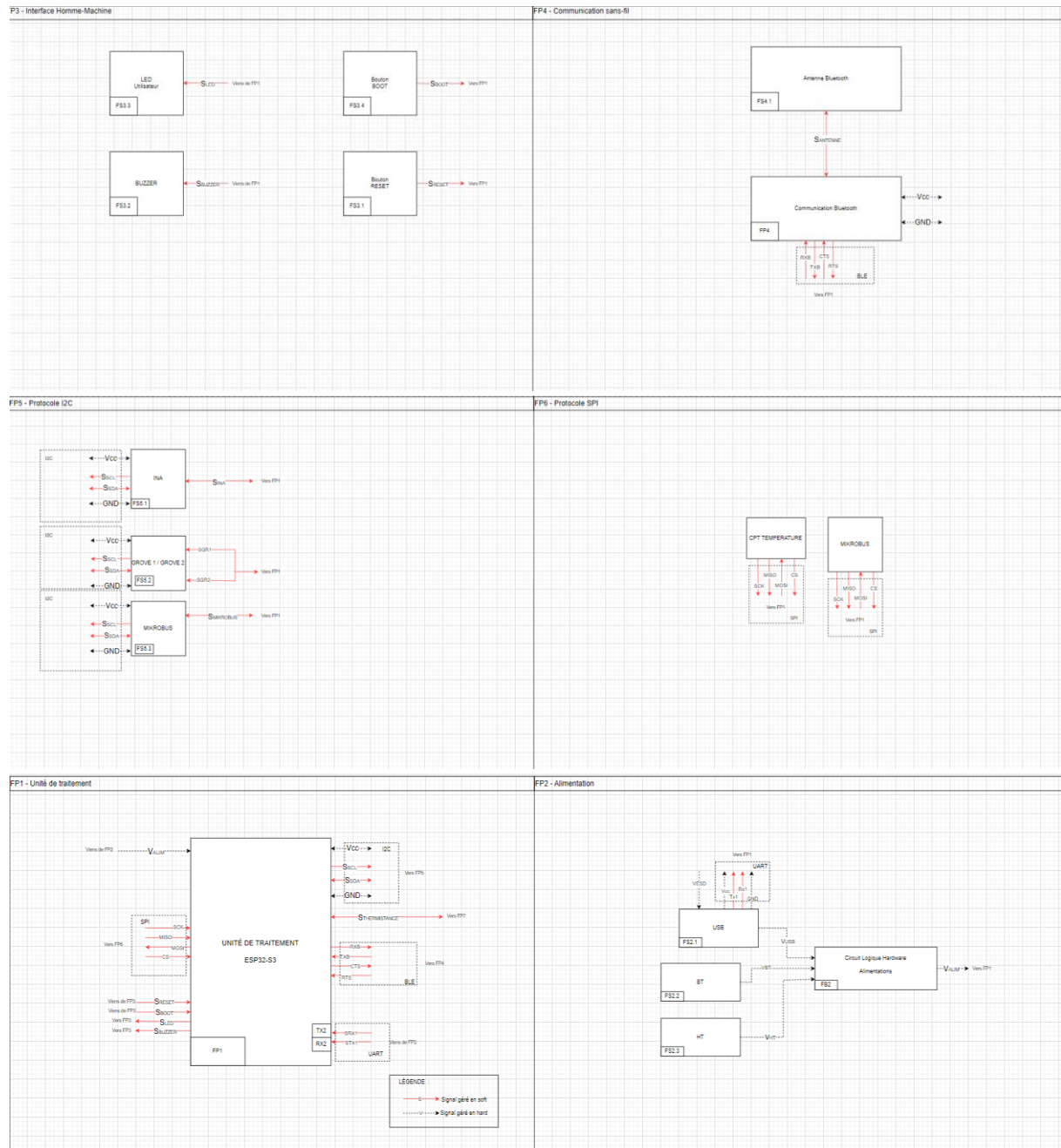
Analyse des Composants

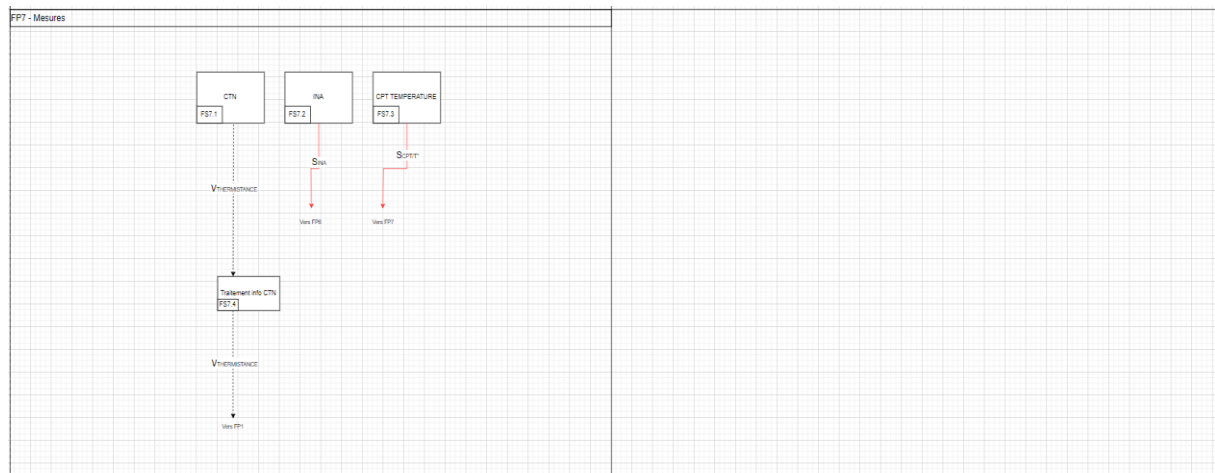
- **Résistances :** Elles sont utilisées pour fixer le point de fonctionnement des amplificateurs opérationnels, pour diviser la tension ou pour limiter le courant.
- **Condensateurs :** Ils sont utilisés pour filtrer le bruit et stabiliser l'alimentation.
- **Microcontrôleur ESP32-PICO-D4** Il est responsable de la communication avec les capteurs, du traitement des données et de la prise de décisions en fonction des mesures.

Annalyse Fonctionnelle 1:



Annalyse Fonctionnelle 2 :





Problématiques et Solutions :

1/ difficulté rencontrée a été l'identification de la fonction des trois broches (pins) de l'antenne. Les informations précises concernant ces broches n'étaient pas facilement accessibles dans la documentation technique. Après une analyse approfondie et des recherches complémentaires, il a été déterminé que :

- La broche 4 est la sortie RF de l'antenne, par laquelle le signal est transmis au circuit suivant.
- La broche 3 et la broche 2 sont connectées à la masse (GND) pour assurer une référence électrique stable et optimiser les performances de l'antenne.

2/ Difficulté par rapport au routage on n'a pas réussi à avoir une carte hygronome

3 / Difficulté à trouver les bonnes datasheets des composants et de calculer les valeurs des composants utilisé