

Conception Électronique ROBOT M1

RAPPORT PROJET CARTE DE DÉVELOPPEMENT ESP32



ALTIUM
DESIGNER

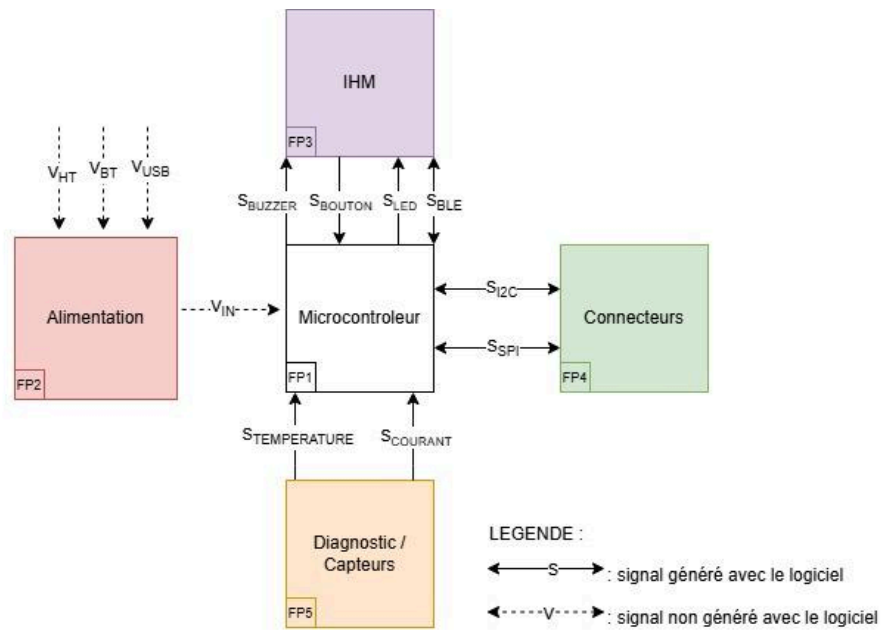
Table des matières

Répartition des tâches dans l'équipe.....	3
AF1.....	4
AF2.....	5
Etude de système.....	6
Sélection des composants.....	8
Bibliothèque.....	10

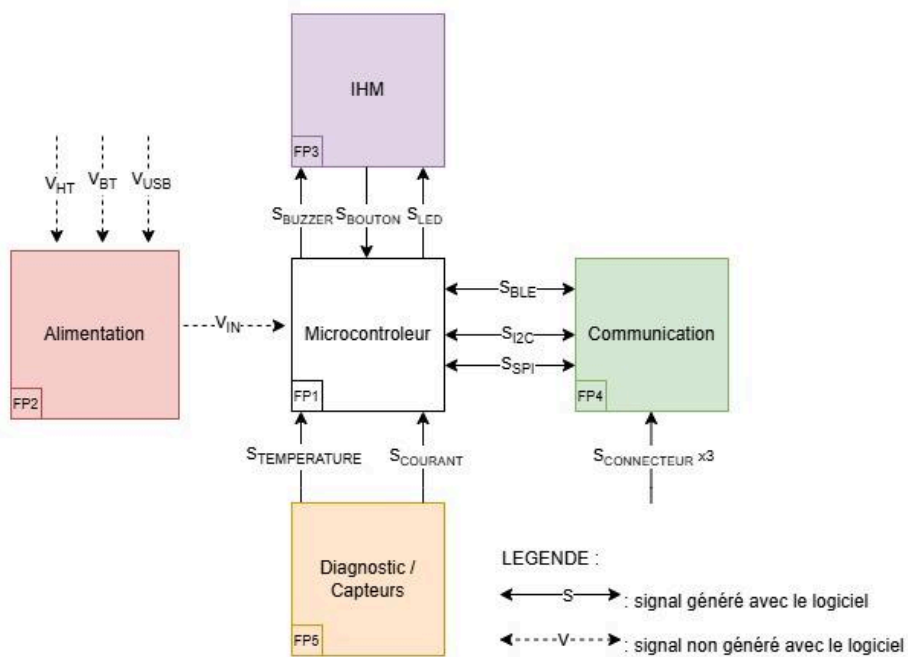
Répartition des tâches dans l'équipe

<u>William</u>	<u>Robin</u>	<u>Léo-Paul</u>
<ul style="list-style-type: none">- Microcontrôleur- IHM (<i>Buzzer, boutons, LED</i>)- Diagnostic/mesures (<i>INA, CTN, temp</i>)	<ul style="list-style-type: none">- Connecteurs (<i>GROVE, Mikrobus</i>)- Driver USB-UART- Antenne BLE	<ul style="list-style-type: none">- Alimentation (<i>HT, BT, USB</i>)- Capteur de mesure courant

AF1

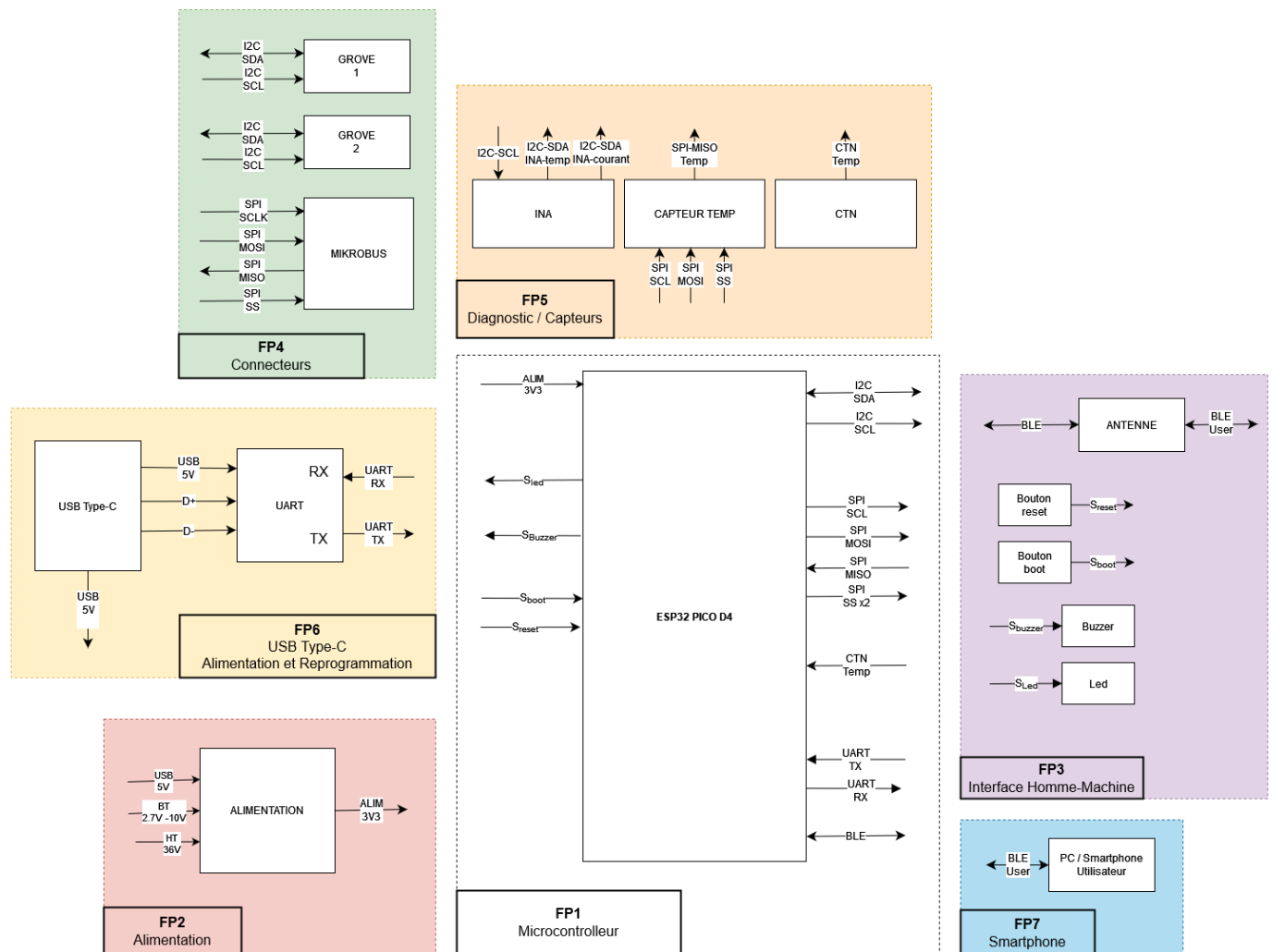


V 1.1



V 1.0

AF2 :



V 1.2

Etude de système :

Objectif du projet : Création d'une carte de développement multi-usage dotée d'une conception modulaire et flexible pour réaliser divers projets / tests.

Le microcontrôleur ESP32 PICO D4 : Il est un module System-in-Package (SiP) qui combine des fonctionnalités Wi-Fi et Bluetooth dans un format compact. Il agit comme le cerveau du dispositif, permettant de contrôler et de gérer divers capteurs, actionneurs et interfaces. Grâce à sa capacité à exécuter des instructions programmées, ce microcontrôleur peut traiter des données en temps réel.

Alimentation permissive : On intègre des options d'alimentation variées (batterie (BT), USB, alimentation secteur (HT)) pour garantir une utilisation dans différents environnements. Les sources d'alimentation peuvent être branchées simultanément et un ordre de priorité permet de ne garder qu'une seule source de tension active.

Capteurs de courant et de température : Ces capteurs permettent de surveiller les performances et de protéger les dispositifs contre les surcharges ou les surchauffes :

- L'INA donne la température du composant.
- Le capteur de température donne la température ambiante.
- Le CTN donne, par une interprétation linéaire, la température du PCB.

Interface Homme-Machine (IHM) : L'IHM comprend un buzzer, une LED, et une connectivité Bluetooth Low Energy (BLE) avec antenne, qui simplifie l'interaction avec l'utilisateur.

Gestion de la carte : On intègre des boutons physiques pour un reset et un boot facilitant le débogage et la maintenance.

Compatibilité avec MikroBus et Grove : Les modules MikroBus permettent d'ajouter facilement des capteurs ou des actionneurs supplémentaires. Les modules Grove offrent une large gamme de composants plug-and-play, facilitant le prototypage rapide. Les modules MikroBus et Grove utilisent principalement les protocoles I2C et SPI pour la communication.

Le protocole I2C (Inter-Integrated Circuit) : Il permet de connecter plusieurs dispositifs sur deux fils (SDA pour les données et SCL pour l'horloge). Il est idéal pour des capteurs qui nécessitent une communication à faible vitesse et qui peuvent partager le même bus, ce qui réduit le câblage.

Le protocole SPI (Serial Peripheral Interface) : Il utilise plusieurs fils (MISO, MOSI, SCK, et un fil de sélection de dispositif) pour une communication plus rapide. Cela le rend adapté aux capteurs ou actionneurs nécessitant un débit de données élevé.

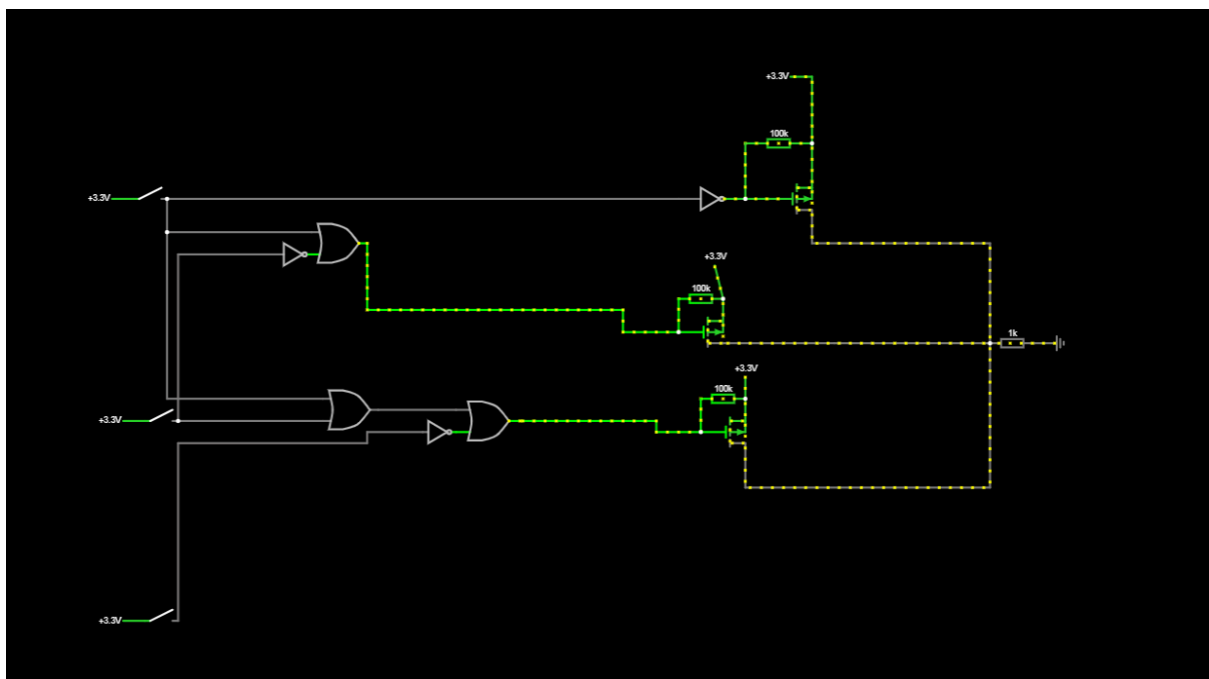
Driver USB-UART : Ce driver permet de convertir les signaux USB en signaux UART, facilitant ainsi la communication entre un microcontrôleur et un ordinateur.

PSRAM : Ajout d'un module de stockage PSRAM de 8Mb afin d'augmenter la capacité limitée de 4Mb du microcontrôleur.

Applications potentielles : Le prototypage rapide pour l'IoT, où des modules peuvent être ajoutés pour tester des concepts rapidement. Les systèmes embarqués nécessitent une surveillance et un contrôle en temps réel, comme les systèmes industriels ou les dispositifs de surveillance environnementale. Des projets éducatifs pour servir de plateforme pour l'apprentissage de l'électronique et de la programmation.

Diodes Schottky pour prioriser les alimentations

Convertisseur buck pour abaisser la tension.



$$\underline{S} = A + C + \underline{B}$$

$$S = \underline{A}.B.\underline{C}$$

$$S = \underline{A}.\underline{C}.B$$

Le choix des tensions se fait grâce au MOSFET P, ainsi qu'aux diodes schottky de préférence qui ont un meilleur rendement et une plus faible tension de commutation.

Ajout de régulateur de tension sur chaque entrée d'alimentation du circuit pour n'utiliser qu'une seule tension (3.3v) dans le circuit logique pour définir quelle source sera utilisé

MCP16301/H Régulateur de tension

Sélection des composants

Microcontrôleur : ESP32-PICO-D4

IHM :

Boutons reset et boot : FSM4JSMATR (Interrupteur tactile) et FSMSM (Interrupteur à bascule).
On prend le FSM4JSMATR pour sa compacité qui permet d'optimiser l'espace de la carte.

LED :

598-8610-207F : LED bi-couleur (rouge/vert), tension directe : 2V et courant direct : 20 mA.

Buzzer :

CMT-8503-85B-SMT-TR (Buzzer électromagnétique) et PKLCS1212E40A1-R1 (Sondeur piézoélectrique).

On prend le CMT-8503-85B-SMT-TR pour sa compacité et pour sa tension de fonctionnement à 3.6 V (celle du PKLCS1212E40A1-R1 est de 12V en général).

Antenne BLE :

NN01-102(Antenne à puce) et NN03-320 (Antenne multiprotocoles).

On prend la NN03-320 car la NN01-102 n'est plus produite.

Choix du MOSFET :

FDV304P (canal P), 2N7002KT1G (canal N), SI3139K-TP (canal P), BSS123LT1G (canal N) et BSS123 (canal N).

Dans la documentation de notre buzzer CMT-8503-85B-SMT-TR on nous conseille le transistor 2SC1741AS on doit donc lui trouver le meilleur remplaçant parmi ce qu'on nous propose. Il est aussi précisé qu'on veut un $V_{ce} < 0.15$ V. On va prendre un canal P pour avoir un Mosfet proche de celui conseillé et on choisit le SI3139K-TP pour sa documentation plus grande.

Alimentation :

Diodes Schottky (1N5819) : Pour éviter que le courant ne retourne vers une source d'alimentation non prioritaire.

P MOSFETs (IRLZ44N ou Si2302) : Pour commuter proprement les sources selon la priorité avec des inverseurs et des portes logiques.

Régulateur de tension (LM7805 ou AMS1117) : Si les différentes alimentations ne fournissent pas la même tension, un régulateur abaisse la tension à une valeur fixe (par exemple 5V).

Les 3 sources seront mises en parallèle avec des diodes sur chacune pour éviter les retours de courant ainsi qu'une résistance à 0 Ohm.

Cette alimentation permet d'alimenter le 3.3v utilisé pour la logique.

Chacune des 3 sources (USB, HT et BT) sont régulées afin d'arriver en 3V3 dans la partie logique de l'alimentation de la carte qui permet le choix de la source d'alimentation.

Une fois la logique respectée, la carte peut-être correctement alimentée.

Composants principaux

Convertisseurs logiques (B1, B3, B4) : MC74VHC1GT32DFT1G : Ce sont des portes OR logiques. Fonction : Elles génèrent un signal d'activation en fonction de la présence d'une tension sur les différentes entrées (USB-C, HT, BT).

Porte logique principale (B2) : 74VHCT05FT : C'est un inverseur avec sortie open-drain. Fonction : Elle coordonne le fonctionnement des MOSFETs en inversant ou en combinant les signaux d'entrée pour assurer la priorité entre les alimentations.

MOSFETs P-Channel (Q3, Q4, Q5) : SI3139K-TP : Ce sont des transistors MOSFET de type P, utilisés pour commuter l'alimentation vers la sortie. Fonction : Chaque MOSFET est associé à une source d'alimentation (USB-C, HT, BT) et laisse passer la tension d'entrée vers la sortie 3V3 si son signal de commande est activé.

Fonctionnement

Le circuit utilise l'alimentation provenant de l'USB-C si disponible, sinon on cherche à utiliser l'alimentation haute tension. Autrement, en dernier recours, on utilise l'alimentation de basse tension.

Sélection des sources : Portes OR (B1, B3, B4) : Elles détectent la présence d'une tension sur les sources USB-C, HT, et BT.

B2 (inverseur open-drain) : Il combine les signaux de sortie des portes OR pour commander les MOSFETs. Les sorties de B2 contrôlent les gates des MOSFETs P-Channel.

Commande des MOSFETs (Q3, Q4, Q5) : Un MOSFET P-Channel est activé lorsque la tension sur sa gate est suffisamment basse (proche de GND). Si une source est priorisée, son MOSFET correspondant sera activé, permettant à cette source d'alimenter la sortie 3V3.

Détail de la logique d'alimentation

1. Présence de **3V3 USB-C** : La sortie de B1 active **Q3 (MOSFET pour USB-C)**. Q3 connecte la source **3V3 USB-C** à la sortie **3V3**.
2. Absence de **USB-C**, mais présence de **3V3 HT** : La sortie de B2 active **Q4 (MOSFET pour HT)**. Q4 connecte la source **3V3 HT** à la sortie **3V3**.
3. Absence de **USB-C** et **HT**, mais présence de **3V3 BT** : La sortie de B3 active **Q5 (MOSFET pour BT)**. Q5 connecte la source **3V3 BT** à la sortie **3V3**.

Le circuit utilise des **portes logiques OU**, un **inverseur open-drain**, et des **MOSFETs P-Channel** pour sélectionner automatiquement la meilleure source d'alimentation disponible. La priorité est gérée de manière logique en activant le **MOSFET** correspondant à la source prioritaire. La sortie **3V3** est ainsi alimentée en continu sans chevauchement entre les sources.

On utilise des **MOSFET P**, car le MOSFET conduit (est activé) lorsque la **gate** est à un potentiel plus faible que la **source**. Le transistor MOS à canal P se commande avec une tension VGS négative ($V_{GS} < 0$) et le courant circule de la source vers le drain : ID négatif. Lorsque la tension VGS est nulle, le courant de drain l'est aussi.

Le MOSFET P est plus adapté que le N pour commuter la ligne d'alimentation positive. La gate peut être abaissée à GND pour activer le MOSFET. Cela permet une sélection propre de la source d'alimentation avec les priorités établies.

Le choix du composant 55620303059 (batterie 3.6V) avait pour but d'alimenter la carte sur batterie. Or je me suis rendu compte que ça ne correspondait pas et que la batterie était beaucoup trop faible pour alimenter la carte. Je me suis rabattu sur un connecteur basique acceptant de la BT, que je n'ai pas eu le temps de schématiser.

Problèmes rencontrés

- Problème de conflits de broche :

La pin IO23 de l'ESP32-PICO-D4 est connectée à deux périphériques : la LED (598-8610-207F) et le capteur TMP126DCKR. Ce conflit peut poser problème si ces deux éléments utilisent cette broche de manière incompatible. Le TMP126 est un capteur de température qui communique via le protocole SPI. Si IO23 est partagée pour SPI et contrôle également une LED, cela entraînera des conflits. La solution trouvée est de déplacer la connexion de la LED sur une broche non utilisée de l'ESP32 : on choisit IO32.

La pin IO22 de l'ESP32-PICO-D4 est partagée entre le buzzer (CMT-8503-85B-SMT-TR) et le capteur INA237AIDGST, ce qui peut entraîner un conflit, car ces deux périphériques ont des utilisations différentes : le buzzer nécessite une fréquence de commande PWM ou un état haut/bas et l'INA237 utilise la broche pour les signaux de communication I2C. La meilleure solution consiste à déplacer le buzzer sur une autre broche libre de l'ESP32 qui supporte les sorties numériques et les signaux PWM nécessaires pour piloter le buzzer : on choisit I25.

- Problème de pins SPI :

Lorsque j'ai voulu utiliser les pins servant au protocole SPI pour lier le composant Mikrobus, j'ai été bloqué par le fait qu'il sont déjà utilisés pour échanger avec la PSRAM.

En recoupant les informations de la datasheet et des recherches parallèles, j'ai découvert que 3 instances du protocoles SPI sont disponibles : SPI, HSPI et VSPI.

De plus, ne comprenant pas la description des pins sur la datasheet, j'ai dû chercher les significations des attributs "CS0, D, Q, WP, HD" attribués aux fonctions SPI.

J'ai ainsi compris que D et Q correspondent respectivement à MOSI et MISO et j'ai pu brancher le composant Mikrobus sur une instance SPI différente de la PSRAM.

Bibliothèque

ESP Pico D4

<https://4donline.ihs.com/images/VipMasterIC/IC/ESPS/ESPS-S-A0009556029/ESPS-S-A0009556029-1.pdf?hkey=EF798316E3902B6ED9A73243A3159BB0>

https://www.espressif.com/sites/default/files/documentation/esp32-pico_series_datasheet_en.pdf

https://www.mouser.fr/datasheet/2/891/esp32_pico_d4_datasheet_en-1365829.pdf

SPI

https://www.mouser.fr/datasheet/2/1127/APM_PSRAM_QSPI_APS1604M_3SQR_v2_7_PKG-1954896.pdf

INA

<https://www.ti.com/lit/ds/symlink/ina237.pdf>

CP2104

<https://www.mouser.fr/datasheet/2/368/cp2104-1397921.pdf>

USB4085

https://www.mouser.fr/datasheet/2/837/GCT_usb4085-3240028.pdf

Mikrobus

<https://download.mikroe.com/documents/standards/mikrobus/mikrobus-standard-specification-v200.pdf>