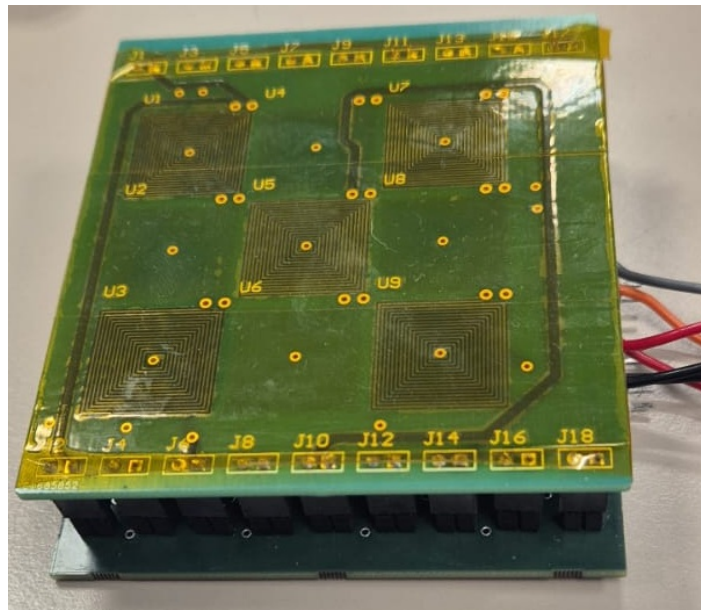


Année académique 24/25
Rapport de projet PSE

Générateur de champs magnétiques bidirectionnels



Option

Systèmes Embarqués

Etudiant

Gaël Mariot

Date

7 mai 2025

Table des matières

1	Introduction	3
1.1	Contraintes	3
2	Théorie	3
2.1	Champ Magnétique	3
2.2	Pilotage bidirectionnels	3
2.3	Architecture	4
3	Choix des composants	4
3.1	Microcontrôleur	4
3.2	Pont en H	5
3.3	LDO	6
3.4	Connecteurs Carte	7
4	Réalisation	7
4.1	Schéma	7
4.2	Layout	8
4.3	Assemblage	8
5	Réalisation du Firmware	12
5.1	Correspondance des pins	12
6	Analyse	13
6.1	Difficultés rencontrées	13
6.1.1	Court-Circuits liés aux aimants	13
6.1.2	Température	13
6.1.3	Oubli du switch	13
6.1.4	LDO trop léger	13
6.2	Consommation	13
6.3	Firmware de test	13
7	Conclusion	14

1 Introduction

Dans le cadre du cours de **Projet en systèmes embarqués**, il nous est demandé de créer un circuit imprimé.

Comme vu en accord avec M. Fourquier, je vais réaliser une carte de pilotage de champ magnétiques pour déplacer des aimants.

1.1 Contraintes

Voici les contraintes de ce projet :

- Utilise un microcontrôleur (*LPC de NXP recommandé*)
- Possède un connecteur de debug (*JTAG ou SWD*)
- Utilise un protocole de communication (*SPI, I2C, etc...*)
- Présence d'un capteur (*Température, Pression, CO2, etc...*)
- Retour utilisateur (*LED, Ecran, Son, etc...*)
- Dimension max de 30mm par 30mm (*Sauf besoin particulier*)
- Peut fonctionner sur batterie

Dans le cadre de mon projet, j'ai été autorisé à ignorer les contraintes de dimensions et du fonctionnement sur batterie.

2 Théorie

2.1 Champ Magnétique

Du courant qui passe dans un fil crée un champ magnétique autour de ce fil dont la polarisation dépend du sens de ce courant.

En créant une bobine, on peut amplifier ce phénomène et obtenir un champ magnétique suffisant pour attirer ou repousser un aimant.

Comme visible sur la Figure 1, le champ magnétique B dépend de du courant I en suivant cette formule :

$$B = \frac{\mu_0 I}{2\pi d}$$

où :

B : La valeur du champ magnétique

μ_0 : La permittivité du vide

I : L'intensité du courant

d : La distance entre le fil et le calcul du champ magnétique

2.2 Pilotage bidirectionnels

Afin d'obtenir un champ magnétique qui puisse avoir 3 états différents (*Attirer, Repousser, Rien*) dépendant du sens du courant parcourant chaque bobine, j'ai dû trouver un moyen d'avoir un circuit qui puisse satisfaire ces états. Ceci peut être réalisé à l'aide d'un pont en H comme visible sur la figure 2.

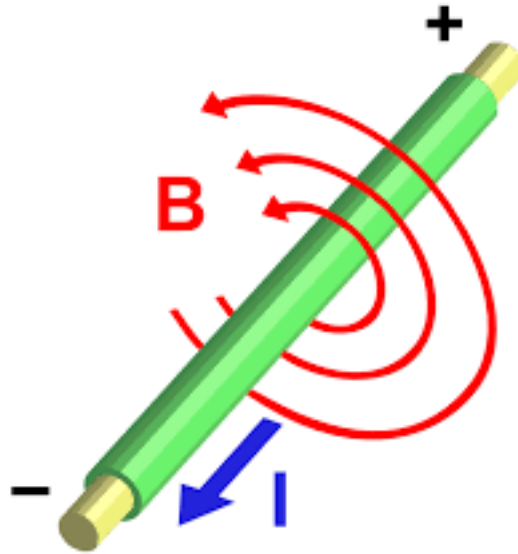


FIGURE 1 – Schéma de fonctionnement d'un champ magnétique

Pour cela, il me faudra 2 entrée pour chaque bobine. Le premier pour définir si un courant parcourt la bobine. Le deuxième pour définir le sens de ce courant.

Il faudra donc un nombre de bits $b = 2n$ avec n étant le nombre de bobines. Dans mon cas je prévois de faire une matrice de 3 par 3 pour fabriquer un Proof of Concept. Il me faudra donc 18 GPIO afin de contrôler les 9 bobines.

2.3 Architecture

Deux cartes seront réalisés dans ce projet. Une carte de contrôle qui contient la logique de fonctionnement avec le microcontrôleur et les pilotes des bobines et une carte de bobines sur lesquels les bobines sont placées.

Comme on peut le voir dans la Figure 3, je prévois de faire un système modulaire qui peut aussi bien être contrôler individuellement via le port UART que par un autre microcontrôleur via un connecteur SPI.

3 Choix des composants

3.1 Microcontrôleur

Voici les contraintes que j'avais

- 18 pins GPIO
- 4 pins SPI
- 2 pins UART
- 3 pins SWD

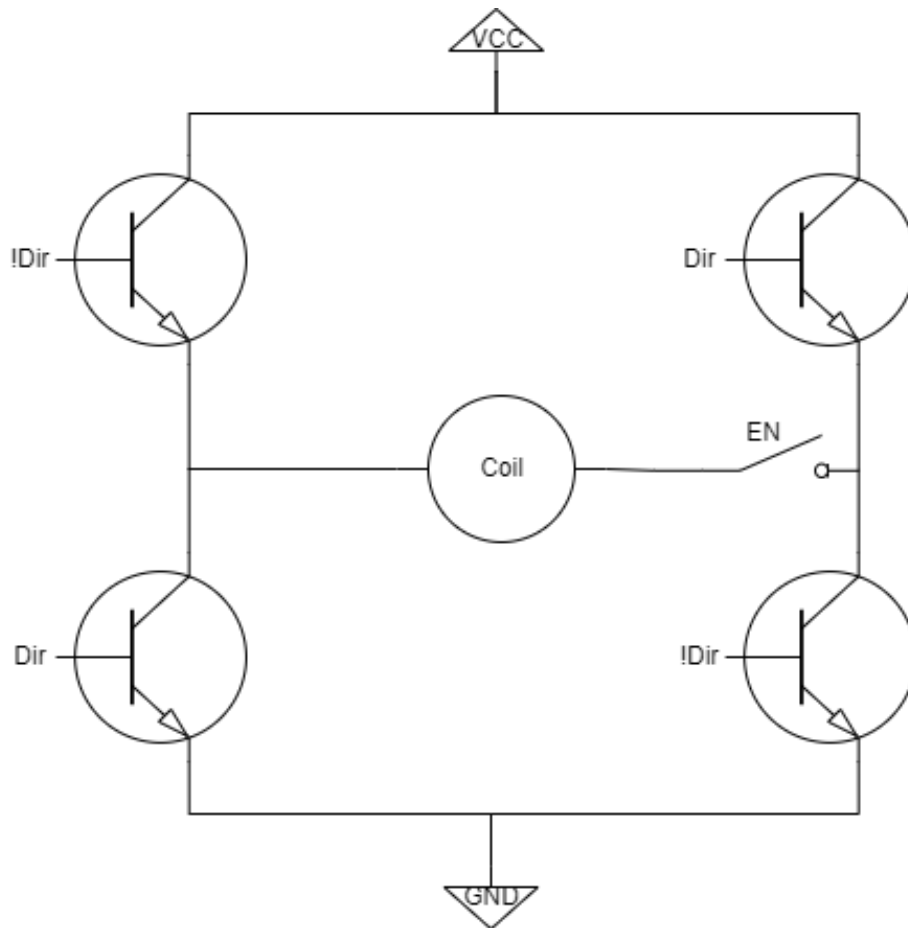


FIGURE 2 – Schéma du pont en H

— 1 pin Reset

Pour un total de 28 pins GPIO nécessaires.

Voici les recherches que j'ai finalement eues

Series	GPIO	CLK max	Conso.	Flash	SPI (M/S)	UART
LPC43xx	49-164	204MHz	100mA	512kB+	51Mb/11Mb	oui
LPC11xx	16-42	50MHz	10mA	4-64kB	25Mb/4Mb	oui
LPC81x	6-18	30MHz	3mA	4-16kB	30Mb/25Mb	oui
LPC804	17-30	15MHZ	1.3mA	32kB	15Mb/20Mb	oui

Mon choix final s'est porté sur le **LPC804M101JHI33E** pour la taille de sa *Flash* plus importante, ses *30 GPIO* et sa *consommation* plus faible que les autres.

3.2 Pont en H

Pour une économie de place, il me faut choisir un pont en H déjà réalisé car utiliser 4 transistor prend plus de place qu'un seul composant.

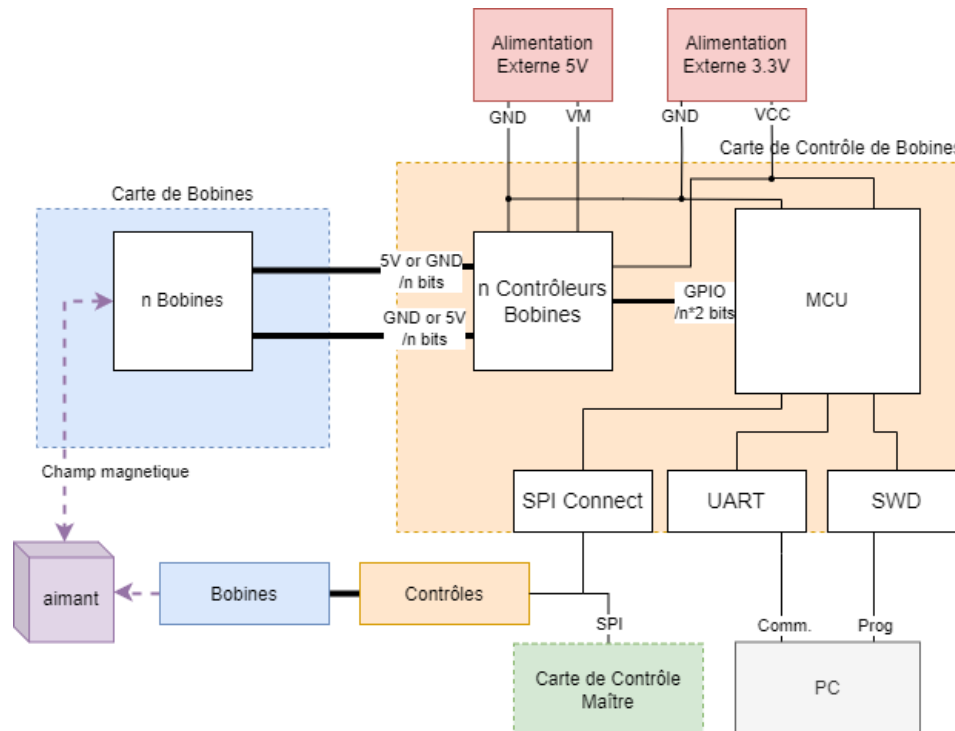


FIGURE 3 – Schéma de l'architecture du projet

Dans un soucis de simplicité, je tenais à avoir le minimum de pin possible et ainsi éviter d'avoir plusieurs Bridge. Pour faire référence au diagramme de la Figure 3, Le bridge se situe dans le bloc "*n Contrôleur Bobines*" et permettrait de contrôler une bobine.

Pour mon choix, j'ai décidé de choisir plusieurs Pont en H. Comme j'ai prévu d'avoir une grille de 3 par 3 bobines, il me faut 9 ponts en H et j'ai décidé d'en prendre 3 différents afin d'omettre tout défaut des composants.

- DRV8213DSGR par Texas Instruments
- TC78H653FTG,EL par Toshiba
- DRV8212DSGR par Texas Instruments

3.3 LDO

Un LDO a été rajouté afin de pouvoir utiliser qu'une seule alimentation 5V à la place d'utiliser une alimentation 3.3V en plus. Comme la consommation en 3.3V est très basse, j'ai choisi un LDO simple qui permet de faire cette conversion.

J'ai porté mon choix final sur le BU33SA5WGWZ-E2 de ROHM Semiconductor pour son prix moins élevé que les concurrents.

3.4 Connecteurs Carte

Il faut prévoir un courant maximum de $5A$ dans le cas où on donne $5V$ en alimentation car les bobines ont une résistance d'environ 1Ω . En doublant les connecteur par fil, le choix des composants augmente car la plupart des connecteurs ont une limite allant de $3A$ à $4.7A$.

J'ai porté mon choix final sur ces composants :

- SSW-102-01-L-S pour les connecteurs femelles
- DW-02-08-G-S-200 pour les connecteurs mâles

La limite du connecteur femelle est de $4.7A$ et le fabricant valide leur utilisation conjointe. La conclusion logique est que le connecteur mâle peut supporter ce courant.

4 Réalisation

4.1 Schéma

J'ai réalisé le schéma de mon PCB avec Altium en utilisant les modèles préconçus par mouser via le *Component Search Engine*

Sur la figure 4, on peut voir les connexions avec tous les H-Bridge avec comme nom *MCxEn* et *MCxPh*. La connexion du SWD, UART et SPI a également été prévue.

Le H-Bridge TC78H653FTG,EL a été connecté comme sur la figure 6 sur les pins (Voir Figure 8) de 1 à 3. Le H-Bridge DRV8212 a été connecté comme sur la figure 5 sur les pins (Voir Figure 8) de 4 à 6. Le H-Bridge DRV8213 a été connecté comme sur la figure 7 sur les pins (Voir Figure 8) de 7 à 9.

Finalement, le deuxième circuit imprimé contient 9 bobines représentée sur la figure 9 et qui sera connecté au premier circuit imprimé via les pins de la figure 8.

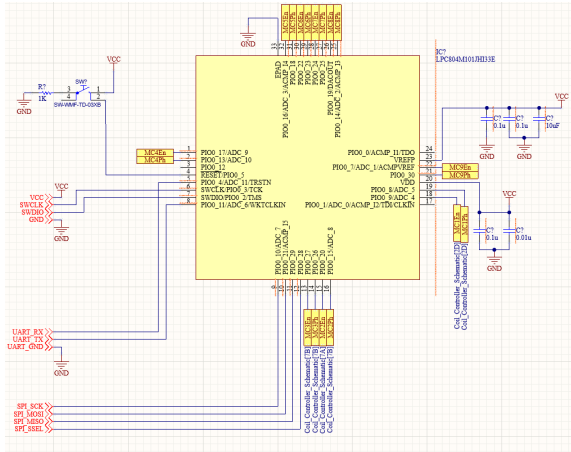


FIGURE 4 – Connection Schématique du LPC804

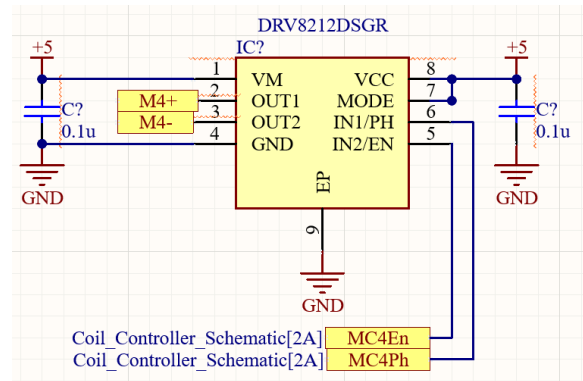


FIGURE 5 – Connection Schématique du DRV8212

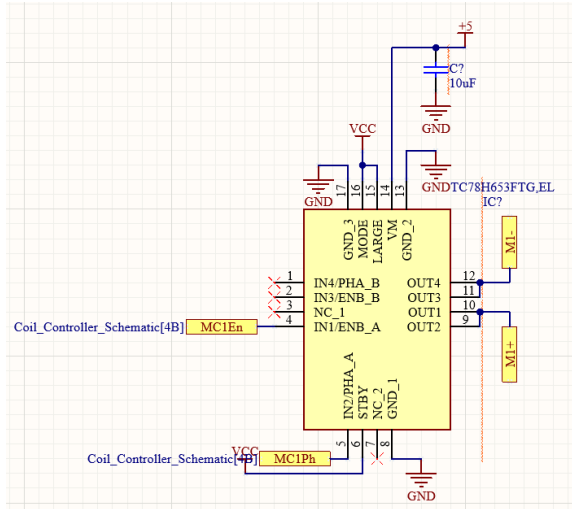


FIGURE 6 – Connection Schématique du TC78

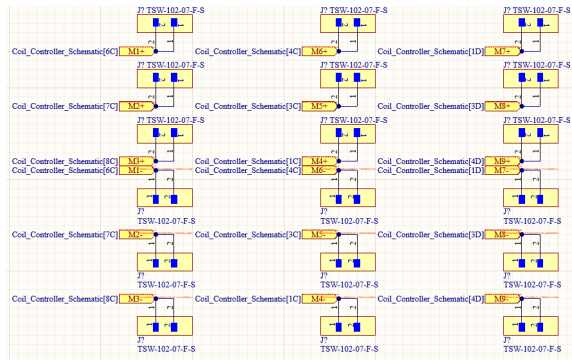


FIGURE 8 – Connection Schématique des Pins

4.2 Layout

Le layout des PCB ont été réalisés sur Altium en s'appuyant sur les librairies fournies par Mouser. J'ai pu réaliser la carte de contrôle (Visible sur la Figure 10) en deux couches mais j'ai dû utiliser 4 couches sur la carte de bobines (Visible sur la Figure 11).

4.3 Assemblage

Pour l'assemblage, j'ai utilisé la Pick'n'Place manuelle et le four vapeur. La pose des connecteurs d'alimentation, UART, SPI et avec la carte de bobines a été réalisé à la main. Le résultat final peut être vu sur les figures 12 et 13 pour la carte de contrôle et sur les figures 14 et 15 pour la carte des bobines.

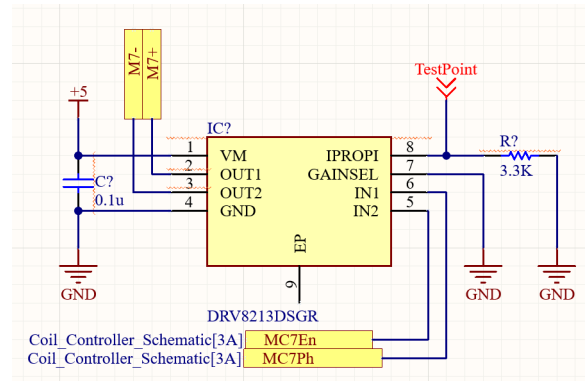


FIGURE 7 – Connection Schématique du DRV8213

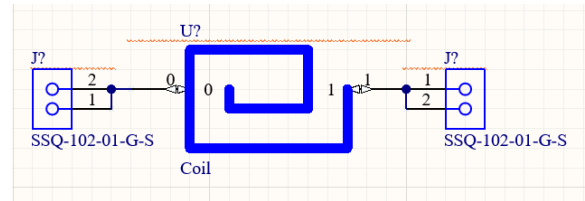


FIGURE 9 – Connection Schématique des bobines

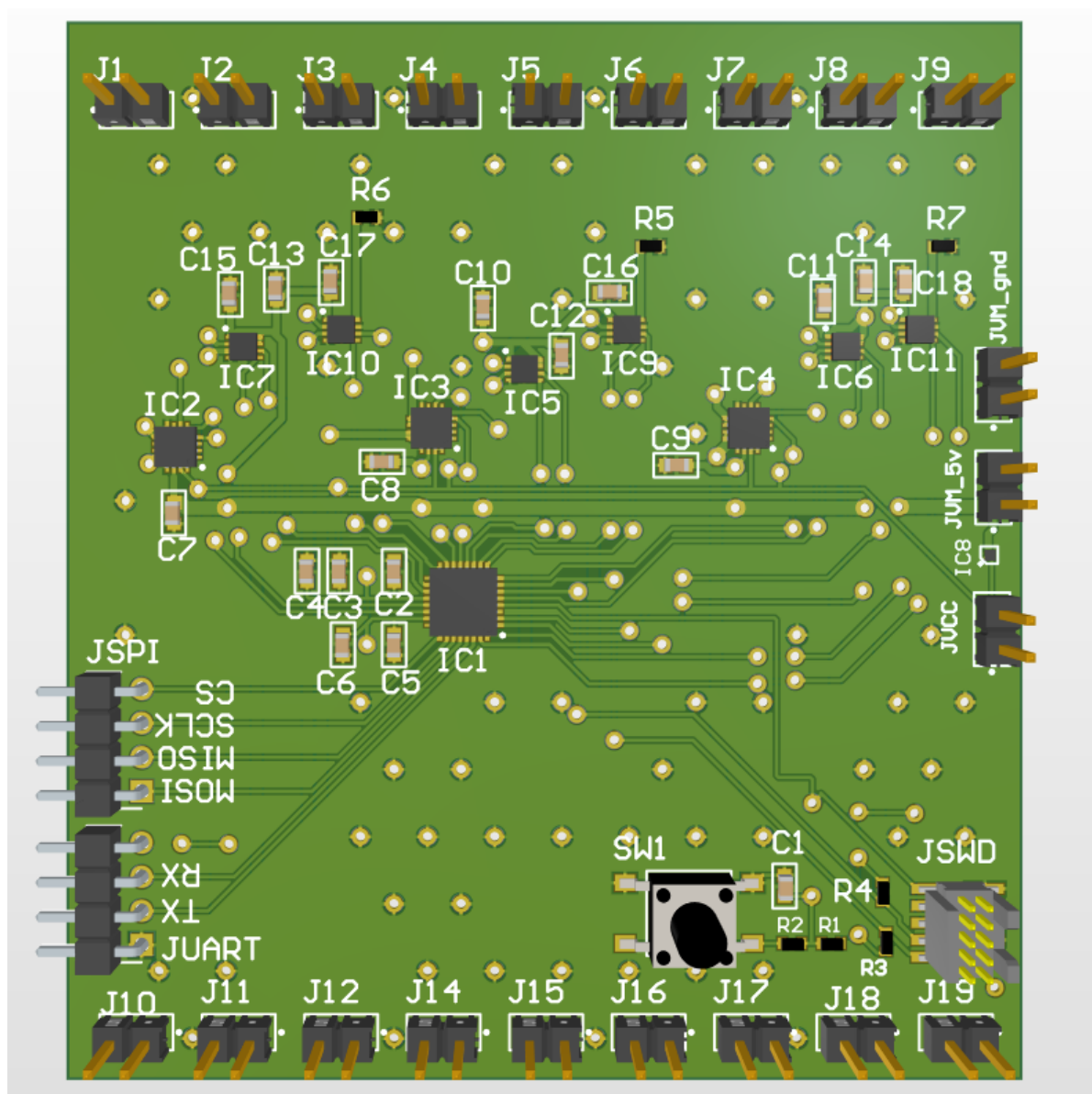


FIGURE 10 – Layout de la Carte de Contrôle sur Altium

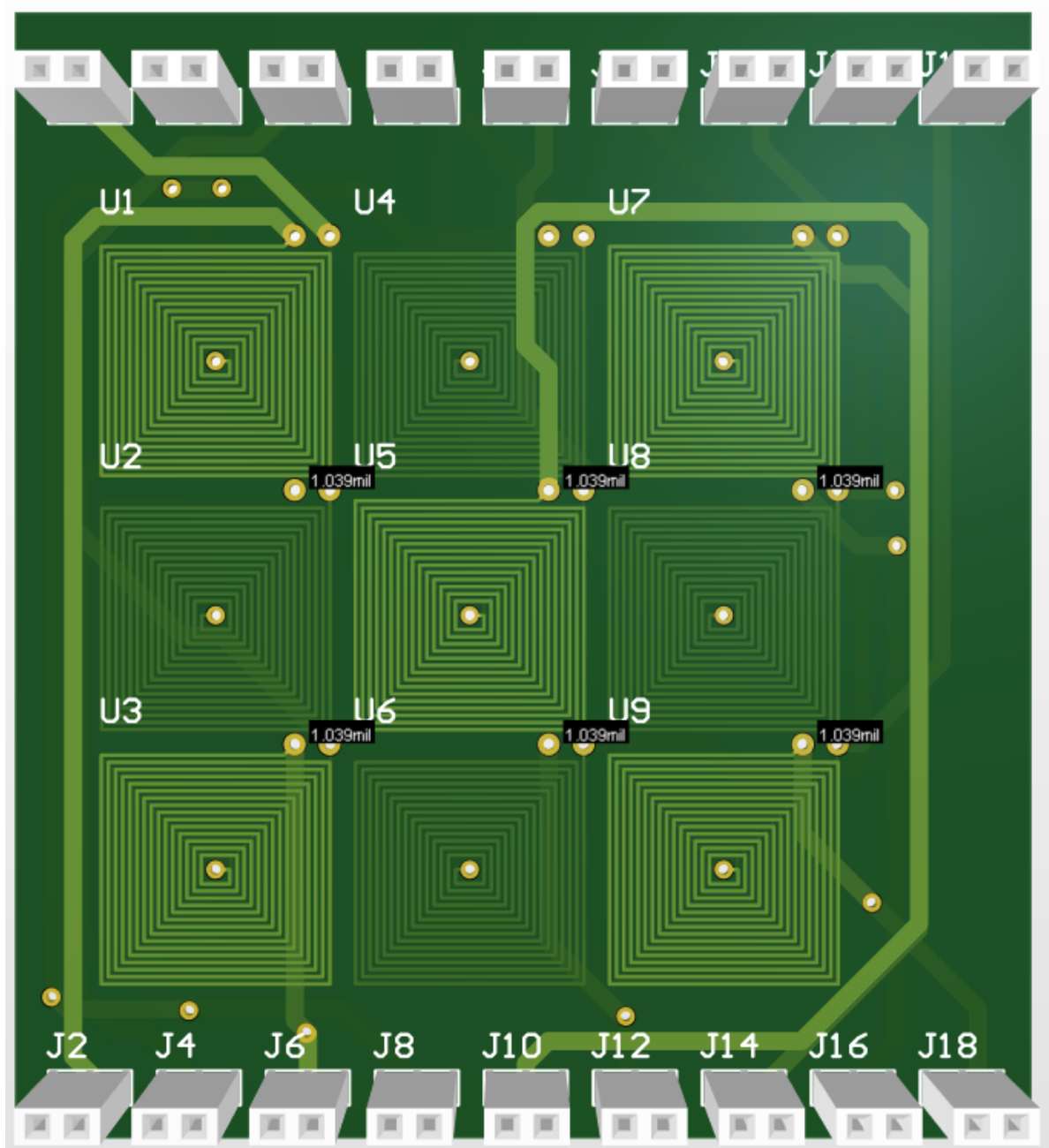


FIGURE 11 – Layout de la Carte des Bobines sur Altium

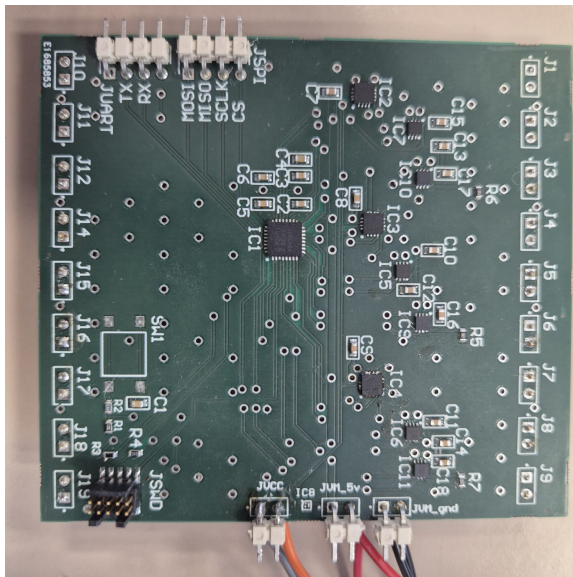


FIGURE 12 – Recto de la carte de contrôle complète et assemblée

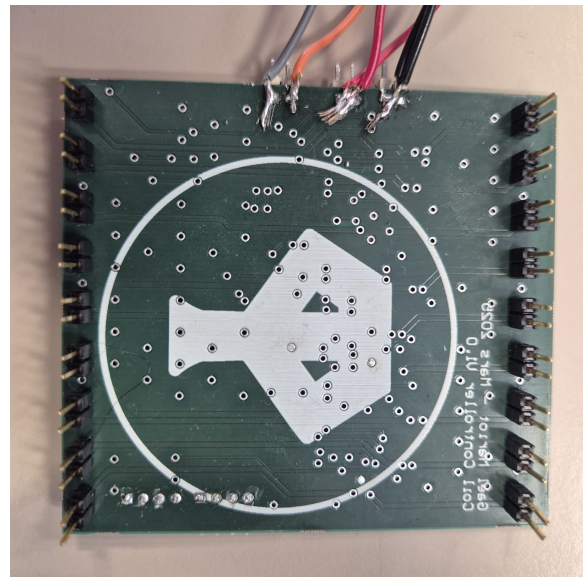


FIGURE 13 – Verso de la carte de contrôle complète et assemblée

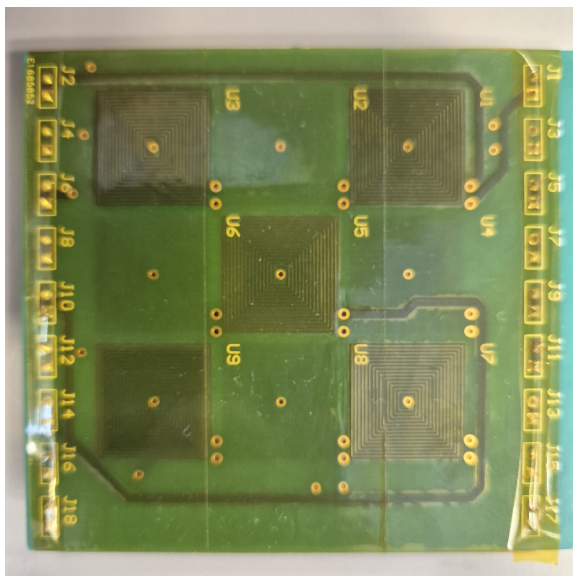


FIGURE 14 – Recto de la carte des bobines complète et assemblée

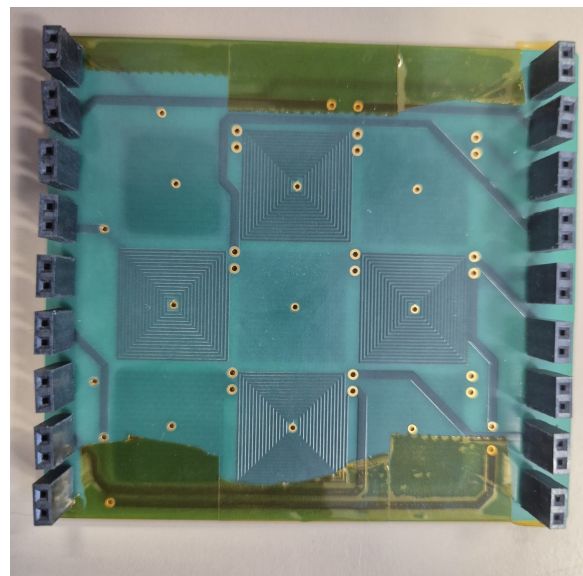


FIGURE 15 – Verso de la carte des bobines complète et assemblée

5 Réalisation du Firmware

Afin de programmer cette carte, j'ai utilisé :

- MCUExpresso v24.12
- Une carte LPC-Link 2

5.1 Correspondance des pins

N° Pin	μC	Connexion Carte
1	<i>PIO0</i> ₁₇	Bobine 8 - Enable
2	<i>PIO0</i> ₁₃	Bobine 8 - Direction
3	Ø	Non Connecté
4	nReset	SWD/Reset
5	<i>PIO0</i> ₄	Bobine 7 - Direction
6	SWCLK	SWD CLK
7	SWDIO	SWD Data IO
8	<i>PIO0</i> ₁₁	Bobine 7 - Enable
9	<i>PIO0</i> ₁₀	Bobine 6 - Avance
10	<i>PIO0</i> ₂₁	Bobine 6 - Recule
11	<i>PIO0</i> ₂₉	Bobine 5 - Direction
12	<i>PIO0</i> ₂₈	Bobine 5 - Enable
13	<i>PIO0</i> ₂₇	Bobine 4 - Direction
14	<i>PIO0</i> ₂₆	Bobine 4 - Enable
15	<i>PIO0</i> ₂₀	Bobine 3 - Avance
16	<i>PIO0</i> ₁₅	Bobine 3 - Recule
17	Ø	Non Connecté
18	<i>PIO0</i> ₉	Bobine 2 - Direction
19	<i>PIO0</i> ₈	Bobine 2 - Enable
20	VDD	VCC
21	<i>PIO0</i> ₃₀	Bobine 1 - Direction
22	<i>PIO0</i> ₇	Bobine 1 - Enable
23	VREFP	VCC
24	Ø	Non connecté
25	<i>PIO0</i> ₁₄	SPI CS
26	<i>PIO0</i> ₁₉	SPI SCLK
27	<i>PIO0</i> ₂₅	SPI MISO
28	<i>PIO0</i> ₂₄	SPI MOSI
29	<i>PIO0</i> ₂₃	UART RX
30	<i>PIO0</i> ₂₂	UART TX
31	<i>PIO0</i> ₁₈	Bobine 9 - Avance
32	<i>PIO0</i> ₁₆	Bobine 9 - Recule

6 Analyse

6.1 Difficultés rencontrées

6.1.1 Court-Circuits liés aux aimants

Mes bobines ayant des vias traversant l'entièreté du PCB et les aimants dont je disposais étant métallique, les aimants provoquaient des courts-circuits lorsque ils étaient disposés sur le PCB. J'ai corrigé cela en posant un film protecteur sur le PCB.

6.1.2 Température

Les bobines ayant un courant élevé passant à travers génèrent beaucoup de chaleur. Il m'est arrivé plusieurs fois de me brûler lors de tests avec 2.9A parcourant les bobines. J'ai dû limiter la limite de courant fournie par l'alimentation à 1A pour que le circuit ne surchauffe pas. Dans un prochain temps, il serait intéressant d'intégrer un radiateur dans la face inférieure de la carte de bobines.

6.1.3 Oubli du switch

Lors de la commande des composants, j'ai omis d'intégrer le switch dans la liste des matériaux. Cependant, ceci n'était pas critique car je pouvais utiliser un câble afin de réaliser le contact.

6.1.4 LDO trop léger

Lors de l'assemblage, le LDO s'est avéré trop léger et a disparu lors du passage au four. Comme il n'était pas nécessaire au fonctionnement de la carte, j'ai connecté une alimentation 3.3V ce qui me permettait en sus de couper l'alimentation des bobines sans couper l'alimentation du microcontrôleur.

6.2 Consommation

La consommation du système est d'environ 5mA sur l'alimentation 3.3V et jusqu'à environ 3A sur l'alimentation 5V. Ceci nous permet d'extrapoler la consommation à un peu moins de 15W.

Les batteries que nous avons étant à 3.7V et sachant qu'elle peut fournir que 2A, on pourrait utiliser la capacité totale de la batterie ce qui la déchargera en 1 heure avec des champs magnétiques moins forts.

6.3 Firmware de test

Afin de tester, un programme a été réalisé afin de tester chaque bobines individuellement en alternant le sens de chaque bobines pour attirer puis repousser chacune à leur tour. Ceci m'a permis de vérifier le fonctionnement correct de mon code ainsi que de vérifier le fonctionnement de tous les pilotes de bobines. J'ai ensuite réalisé un programme

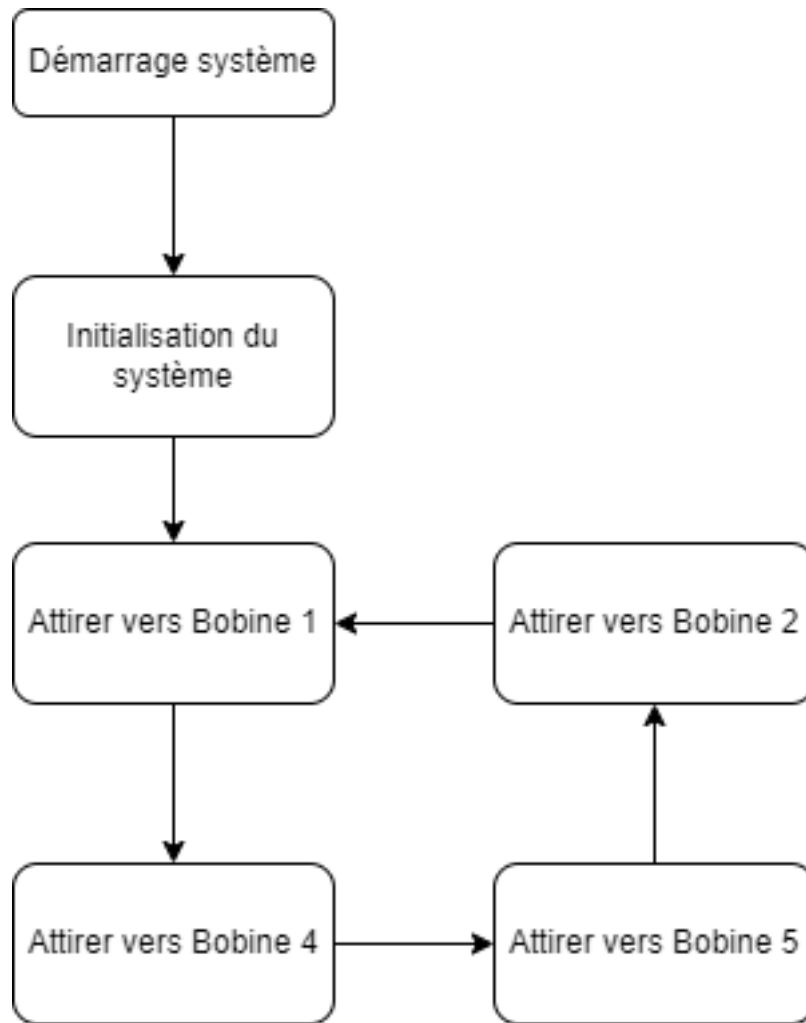


FIGURE 16 – Diagramme de flux du firmware de test

de démo qui alterne entre 4 bobines et qui permet de voir le trajet des aimants sur mon pcb. Le fonctionnement du programme est visible sur la Figure 16

7 Conclusion

Ce projet de générateur de champs magnétiques bidirectionnels a permis de mettre en pratique des concepts théoriques en électronique embarquée, tout en relevant plusieurs défis techniques. Malgré les difficultés rencontrées, les solutions apportées ont conduit à un prototype fonctionnel. La consommation mesurée (15W max) et les tests logiciels confirment la viabilité du système.

Pour des améliorations futures, l'ajout d'un radiateur, l'utilisation de bobines planaires différentes ou d'un agencement différent pourraient améliorer l'efficacité du système.