

Simulations du circuit avec LTSpice

Nous devons concevoir un circuit électronique autour du capteur afin d'optimiser et d'amplifier le signal de sortie, car la tension récupérée à la sortie du capteur est très faible. En effet, la résistance du capteur est très élevée, d'environ 50 M Ω , avec une tension d'alimentation de 5V. Cela entraîne un courant de sortie du capteur d'environ 100 nA, ce qui est insuffisant pour la carte Arduino. Par conséquent, nous devons amplifier ce courant, tout en étant conscients des risques associés.

Avant de créer le circuit imprimé (PCB), il est très utile d'utiliser un logiciel tel que LTSpice pour effectuer des tests sur le circuit. Cela nous permet de vérifier à l'avance que le fonctionnement correspond bien à nos attentes, afin d'éviter d'éventuels problèmes ultérieurs. Nous avons réalisé des simulations du circuit, ainsi que des simulations des solutions mises en place pour une utilisation optimale de l'amplificateur, en plaçant judicieusement des filtres.

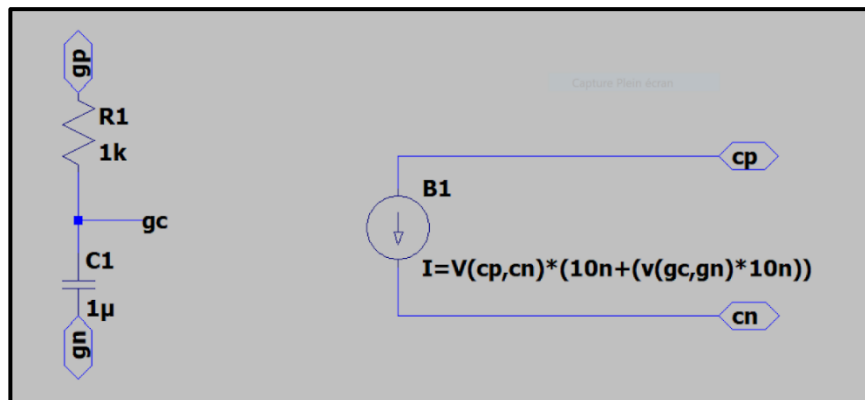


Figure 1 : Capteur modélisé avec LTSpice

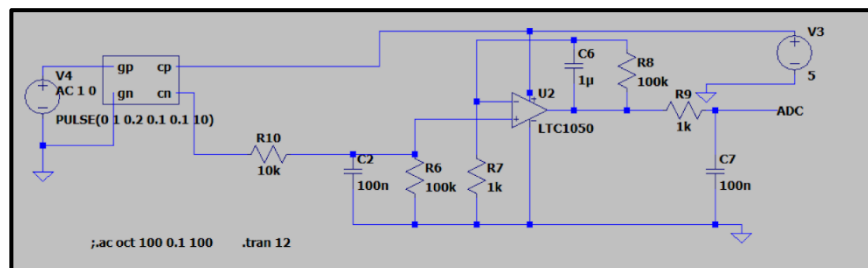


Figure 2 : Circuit dans lequel le capteur est intégré

Simulation dans les conditions nominales

Nous avons effectué une simulation transitoire d'une durée de 12 secondes pour simuler le fonctionnement normal du circuit avec le capteur. L'objectif était de vérifier si le signal était correctement amplifié et dans une plage détectable par la carte Arduino. Dans la Figure 3, nous pouvons observer que la tension de sortie du montage atteint 1V, ce qui confirme sa détectabilité.

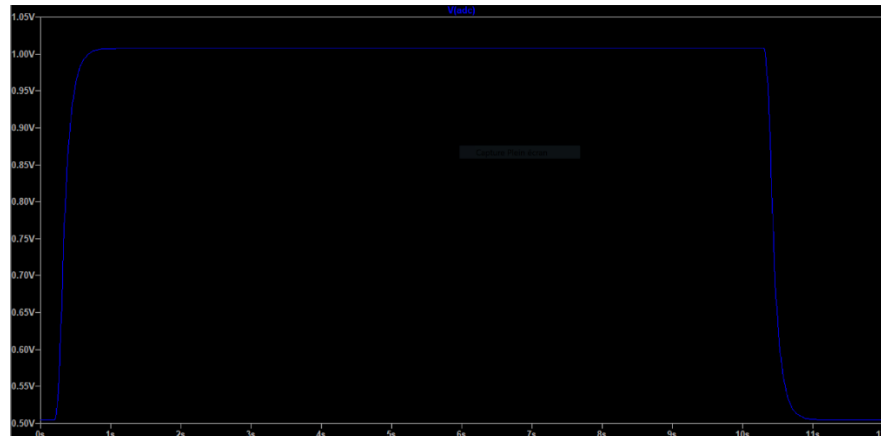


Figure 3 : Simulation du circuit dans les conditions nominales

Caractérisation des filtres

Il est nécessaire d'inclure des filtres pour réduire les bruits indésirables pouvant noyer le signal utile. Nous avons utilisé trois filtres passe-bas, chacun atténuant des sources de bruit différentes. Pour les simulations de chaque filtre, nous avons activé uniquement l'étage de filtre qui nous intéressait à chaque fois, ce qui nous a permis de caractériser chaque filtre avec leurs fréquences de coupure respectives. Les simulations .ac ont été utilisées pour obtenir les diagrammes de Bode de sortie.

Filtre 1- Atténuation du bruit dû à l'alimentation du circuit, avec une fréquence de coupure de 1.6 Hz.

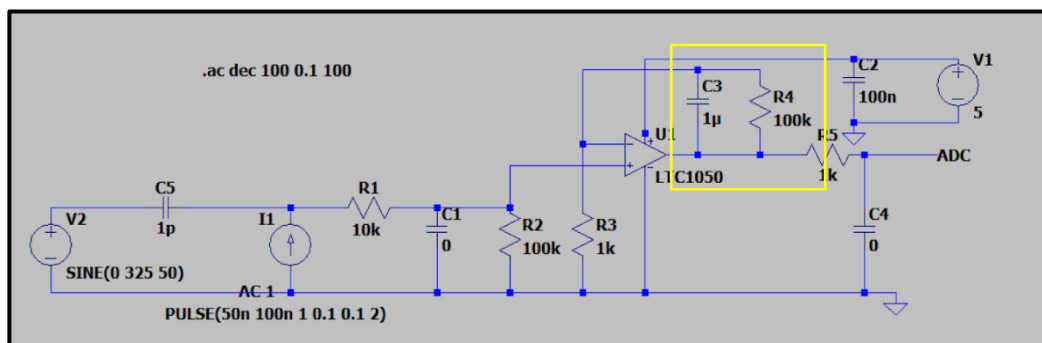


Figure 4 : Circuit utilisé pour la simulation du Filtre 1

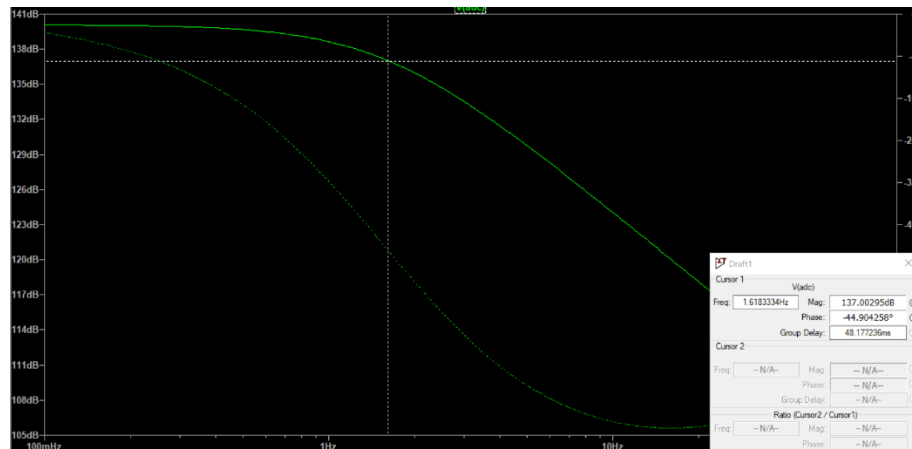


Figure 5 : Tracé de Bode avec le Filtre 1

Filtre 2- Atténuation du bruit du signal récupéré à la sortie du capteur, avec une fréquence de coupure de 16 Hz.

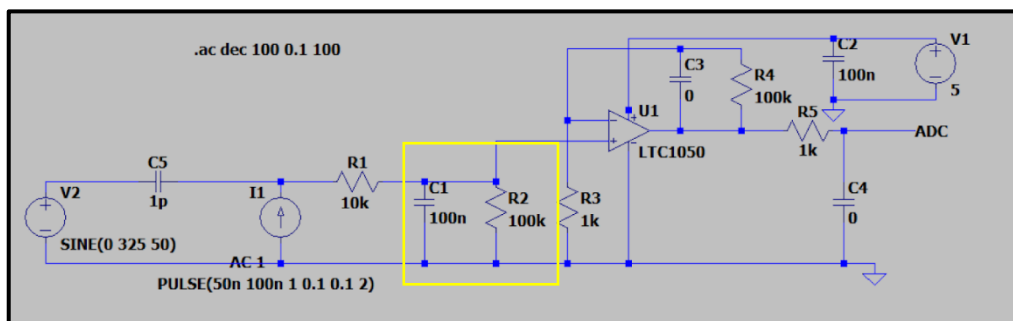


Figure 6 : Circuit utilisé pour la simulation du Filtre 2

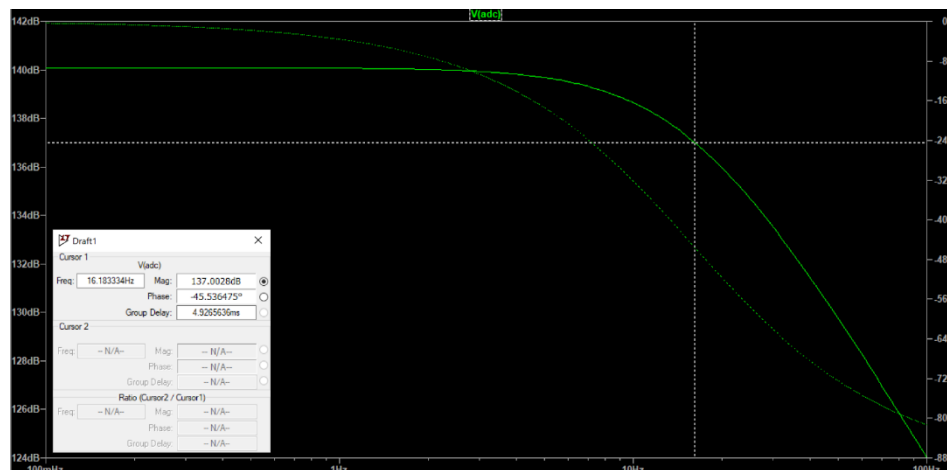


Figure 7 : Tracé de Bode avec le Filtre 2

Filtre 3 : Atténuation du bruit à l'entrée de l'ADC, avec une fréquence de coupure de 1.6 kHz.

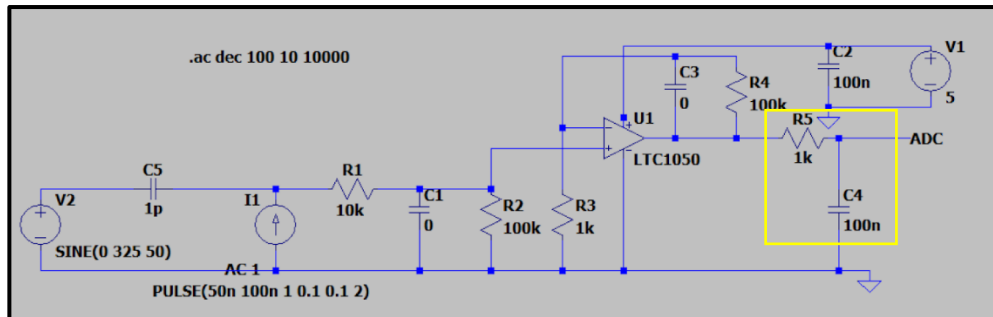


Figure 8 : Circuit utilisé pour la simulation du Filtre 3

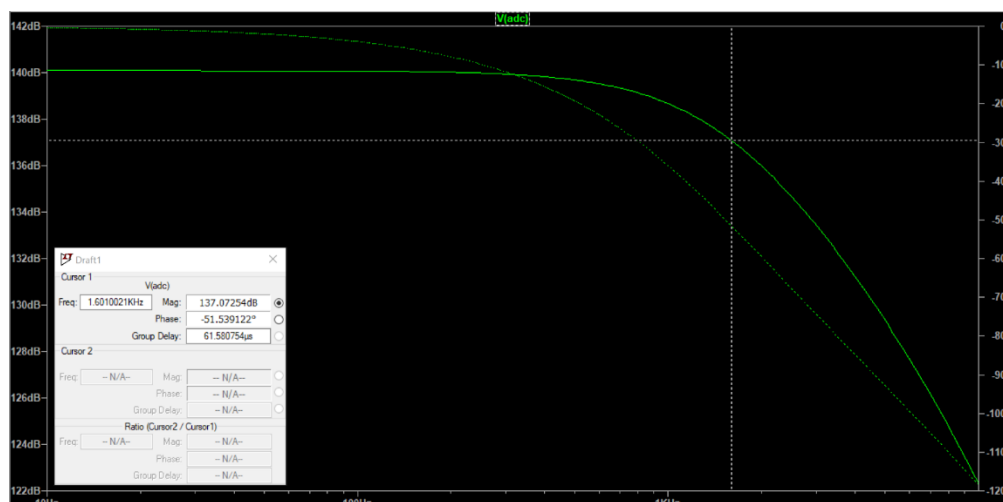


Figure 9 : Tracé de Bode avec le Filtre 3

Evaluation de l'efficacité des filtres

Il est maintenant nécessaire de tester les filtres mis en place dans le montage comportant le capteur. Pour cela, il suffit d'introduire un bruit de 50 Hz représentant des interférences électriques de l'alimentation secteur. Nous réalisons ensuite une simulation en courant alternatif afin de trouver l'atténuation liée à cette fréquence, qui nous permettra de déterminer si les filtres sont efficaces ou non.

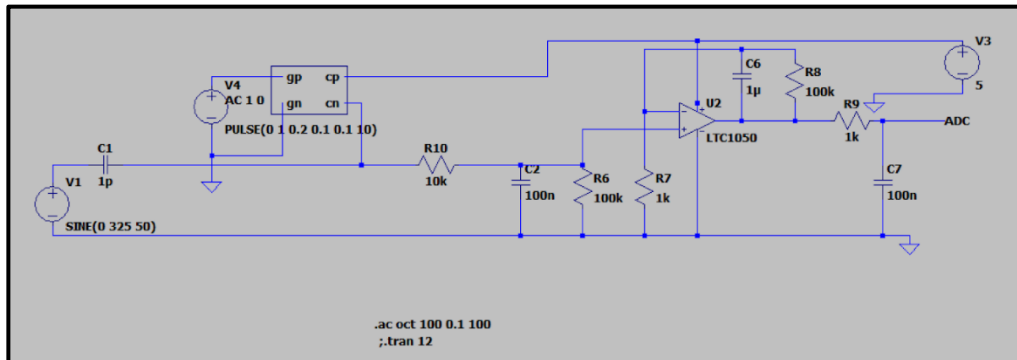


Figure 10 : Circuit avec tous ces filtres utilisé pour l'évaluation de l'efficacité des filtres

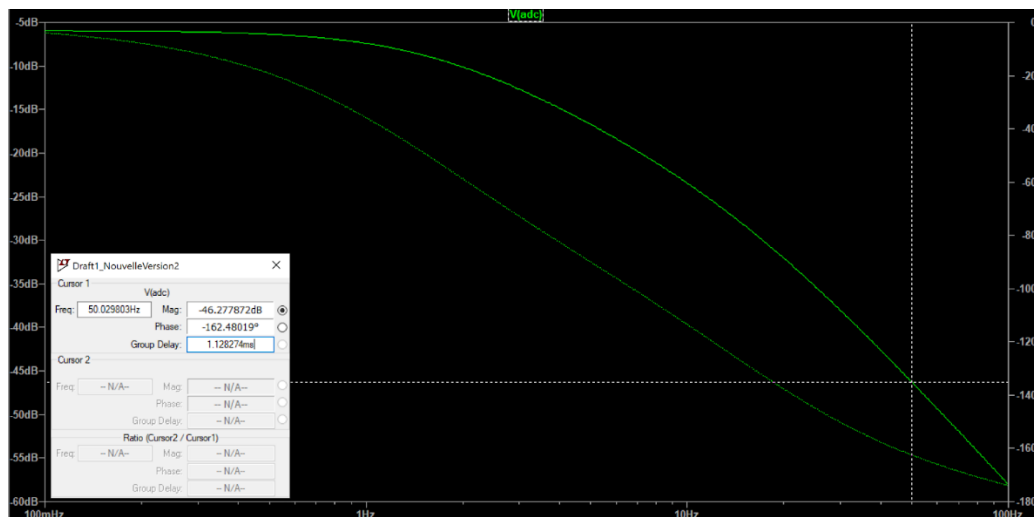


Figure 11 : Diagramme de Bode avec un bruit en courant de 50Hz

À 50 Hz, on obtient un gain de -46 dB, ce qui correspond à une atténuation totale de 39 dB. L'atténuation à la fréquence indésirable étant élevée, nous pouvons alors conclure que les différents filtres sont efficaces.

La fréquence de Nyquist

Enfin, il est nécessaire de vérifier que notre montage est fonctionnel à la fréquence de Nyquist. La fréquence de Nyquist est la fréquence maximale que doit contenir un signal pour permettre sa description sans perte de qualité par un échantillonnage à intervalles réguliers. Elle est de la moitié de la fréquence d'échantillonnage. Sur une carte Arduino Uno, la plus grande fréquence d'échantillonnage utilisable possible est de 15.4 kHz. La fréquence de Nyquist est donc de 7.7 kHz. Nous simulons donc un bruit à cette fréquence, ce qui nous permettra de déterminer l'atténuation pour ce cas.

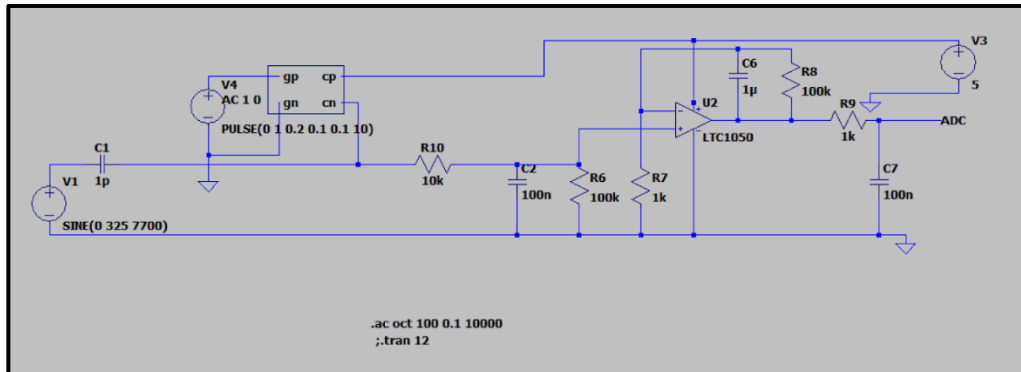


Figure 12 : Circuit complet utilisé pour le calcul de l'atténuation de bruit à la fréquence de Nyquist

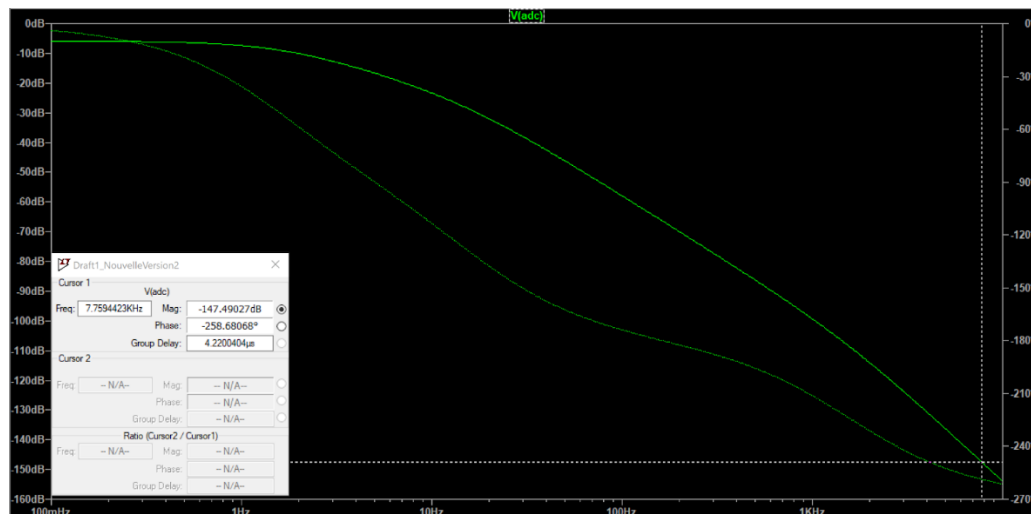


Figure 13 : Diagramme de Bode de la sortie avec du bruit à la fréquence de Nyquist

À partir du diagramme de Bode, nous pouvons observer une atténuation de 140 dB pour un bruit à la fréquence de Nyquist. Cette conclusion est basée sur la comparaison des gains, où nous constatons un gain de -7 dB à 0.1 Hz et un gain de -147 dB à 7.7 kHz. Ces valeurs indiquent que le bruit est complètement atténué, ce qui démontre l'efficacité de notre filtrage.

Conclusion

Les simulations effectuées avec LTSpice ont été essentielles pour caractériser chaque élément du circuit, vérifier le fonctionnement de chaque partie et évaluer les performances globales du système. Elles nous ont permis de valider les choix de conception, d'optimiser les paramètres des filtres et d'assurer que le signal utile était amplifié de manière adéquate.

Maintenant que nous avons validé le concept et le fonctionnement de notre circuit, nous sommes prêts à passer à la prochaine étape, qui consiste à concevoir et réaliser le PCB qui mettra en œuvre l'ensemble des composants et connexions de manière optimale.